Universidad de Alcalá

Escuela Politécnica Superior

Departamento de Electrónica

Generación y correlación eficiente de códigos binarios derivados de conjuntos de secuencias complementarias para sistemas ultrasónicos

Autora: Mª Carmen Pérez Rubio

Directores: Dr. Jesús Ureña Ureña

Dr. Álvaro Hernández Alonso

Tesis doctoral

2009

Resumen

En esta tesis se evalúan distintos esquemas de codificación aplicados a señales ultrasónicas para la determinación precisa de tiempos de vuelo. Una adecuada codificación confiere al sistema de una mayor inmunidad ante el ruido, la capacidad de discriminar varias emisiones realizadas simultáneamente, y además permite conseguir una gran precisión en la medida de distancias. Entre las diversas aplicaciones se encuentran la detección de obstáculos, tareas de localización en sistemas de computación ubicua o ensayos no destructivos en materiales. Otras áreas, como las comunicaciones, criptografía o radar, también demandan el uso de códigos con propiedades de correlación favorables.

La tesis puede dividirse en tres grandes bloques. El primero se dedica al estudio de las propiedades de correlación aperiódica de los códigos binarios actuales más prometedores, y a la propuesta de nuevos esquemas de codificación. El segundo aborda el diseño de generadores y correladores eficientes que reducen en gran medida el número de operaciones a realizar para llevar a cabo la generación o detección de los códigos previos. Finalmente, en el último bloque se analiza la aplicación de los distintos códigos a un sistema de posicionamiento local.

Tras un análisis pormenorizado de los códigos de mayor aplicación, incluidas las últimas propuestas con códigos ortogonales generalizados LS y T-ZCZ, se realiza una selección de aquellos sub-conjuntos de códigos con menores lóbulos laterales de auto-correlación y correlación cruzada aperiódica. Asimismo, se ha comprobado que la correlación aperiódica parcial de los códigos elegidos permite reducir casi en su totalidad la zona ciega de exploración típica de los sistemas sensoriales ultrasónicos que utilizan un mismo transductor tanto en emisión como en recepción.

Se presenta un nuevo algoritmo para la obtención de pares T-ZCZ a partir de un esquema de generación de conjuntos complementarios de secuencias (CSS) de reciente aparición. Estos pares presentan zonas de correlación cero en la suma de sus funciones de correlación aperiódica y permiten salvar las restricciones de los CSS en el número de secuencias asignadas a cada emisor. Los pares T-ZCZ aquí propuestos presentan zonas de interferencias de menor tamaño y valores de cota inferiores en dichas zonas a los descritos en trabajos previos; y, lo que es de gran importancia, permiten obtener generadores y correladores eficientes que minimizan la carga computacional de su implementación hardware, si se

comparan con implementaciones directas. El diseño de ambos módulos, de generación y correlación, se acomete también en la tesis. El primero permite obtener simultáneamente y de forma sencilla todos los pares ortogonales generalizados de una misma familia. Asimismo, el segundo proporciona a su salida la correlación de la señal de entrada con todos los pares de la familia.

Las propiedades de correlación de los CSS, y el hecho de que la mayor parte de códigos ortogonales generalizados deriven de ellos, animan a continuar los esfuerzos investigadores en esta línea. En la tesis se propone una modificación de los algoritmos de generación y correlación asociados a dichos códigos, que da lugar a una implementación genérica en hardware configurable, capaz de operar en tiempo real y de adaptarse a los requisitos de la aplicación concreta. Esta implementación se utiliza como base para la propuesta de nuevos algoritmos de generación y correlación de códigos LS, tanto de los obtenidos a partir de pares Golay como de los obtenidos utilizando CSS. La zona libre de interferencias que estos códigos presentan alrededor del origen, junto con la disponibilidad de los correladores aquí propuestos, hacen muy atractivo su uso en aplicaciones cuasi-síncronas.

Finalmente, se ha definido un sistema de posicionamiento local ultrasónico, en el que se ha podido comprobar experimentalmente la validez de los resultados obtenidos con todos los esquemas de codificación evaluados y algoritmos propuestos.

Abstract

In this thesis, current and new CDMA encoding schemes are evaluated for their application in ultrasonic sensory systems based on the determination of times-of-flight. Proper encoding improves the performance of such systems in terms of noise immunity, capability of simultaneous measurements and precision in the distance measurements. Important applications include encoded ultrasonic signals, such as obstacle detection, local positioning in ubiquitous computing or non-destructive testing. Furthermore, other fields such as radar, cipher cryptology or communications systems, also demand codes with favourable correlation properties.

The thesis can be divided into three major parts. The first looks at the aperiodic correlation properties of the most promising current binary codes, and proposes novel encoding schemes. The second deals with the design of efficient generation and correlation algorithms that notably decrease the number of operations necessary for the generation or detection of the codes. Finally, in the last part, the application of the codes in a local positioning system is discussed.

After a detailed analysis of various binary codes, including the generalized orthogonal LS and T-ZCZ recently discovered, a selection of those subsets with lower aperiodic auto-correlation sidelobes and aperiodic cross correlation values is given. Also, the partial aperiodic correlation properties of the chosen codes nearly eliminates the blind area that appears in ultrasonic sensory systems when the same transducer works as emitter and as receiver.

An important contribution presented in this thesis is a new generation algorithm of T-ZCZ pairs derived from a construction of complementary sets of sequences (CSS) recently proposed. These pairs have three zero correlation zones in the sum of their aperiodic correlation functions, and they overcome the restrictions of CSS in the number of sequences assigned to every emitter. The proposed T-ZCZ codes have smaller zones with interferences and lower maximum bounds in these zones than previously proposed generation schemes for T-ZCZ codes. And most importantly, these new codes can be very efficiently generated and correlated, if compared to a straightforward implementation. The design of the efficient generation and correlation algorithms is also presented. The first one, allows for an easy simultaneous generation of all the generalized pairs in a family. The second one

simultaneously performs the correlation of the input signal with each pair of the family.

The ideal correlation properties of CSS, and the fact that most generalized orthogonal codes are derived from them, suggest that more effort has to be invested in these codes. This thesis proposes a modification of the efficient generation and correlation algorithms of these codes to achieve a generic implementation on a configurable architecture, capable to operate in real time and to be adapted to requirements from different applications. This implementation is the basis of novel algorithms proposed for the efficient generation and correlation of LS codes, generated either from Golay pairs or from CSS. This fast correlator, together with the zero correlation zone that LS codes exhibit, make them a good choice for quasi-synchronous applications.

Finally, the definition of an ultrasonic local positioning system has been explained. It makes it possible to verify the performance of the various encoding schemes used in the thesis, as well as the proposed efficient algorithms.

A Miguel Ángel

Agradecimientos

Me gustaría dar las gracias en primer lugar a mis directores de tesis, Jesús Ureña y Álvaro Hernández, a los que debo más que un compromiso intenso en la dirección de esta tesis. A nadie que haya tenido ocasión de trabajar con ellos le sorprenderá que mencione su entrega a la docencia y a la investigación, el impulso y apoyo que sin reservas son capaces de brindar a alumnos y compañeros, su buen hacer y admirable generosidad.

Del mismo modo me gustaría agradecer los consejos y orientaciones de Carlos de Marziani y Fernando Álvarez. El trabajo realizado en esta tesis se nutre en parte del esfuerzo que ellos realizaron en las suyas. No puedo menos que darles las gracias por todas las dudas resueltas y, sobre todo, por su disponibilidad más allá de cualquier horario.

También fuera de España tengo a quien dar las gracias. A Liam Marnane, por su ayuda desinteresada, sus ideas para afrontar las tareas de implementación, por iniciarme en el mundo del LaTeX y por el cálido recibimiento que él y su familia nos dieron a Miguel y a mi cuando llegamos a Irlanda. Asimismo, de mi estancia en Bélgica debo dar las gracias a Herbert Peremans y al resto de chicos del laboratorio, por enseñarme su trabajo, su calidad humana y por hacer que mereciese la pena el tiempo pasado fuera de casa.

Mis deudas no acaban ahí. A Isaac y Dani les debo la programación de los algoritmos de alto nivel de la etapa de posicionamiento. Tengo que agradecer también la ayuda y palabras de aliento de Ana y Juan Jesús. No me olvido de mis compañeros de laboratorio, por proponer ideas, por contribuir a aligerar el peso de la tesis en cada descanso, pero sobre todo por crear un ambiente de trabajo difícil de superar.

Hago extensivo mi agradecimiento a todo el Departamento de Electrónica de la UAH, por haber ayudado con consejos y buenos ratos a hacer más llevadera la tesis.

En el terreno de lo personal, los primeros agradecimientos van para mis padres, que han mostrado siempre interés por este trabajo, dándome ánimo, apoyo constante y perdonando mis escasas visitas. A mi hermano por estar cuando lo necesito. A la familia de Miguel por el apoyo logístico, dispuestos siempre a echar una mano en casa para que no hubiese distracciones que me separasen de la tesis. A mis amigos, que siguen llamándome a pesar de mis ausencias. Algún imprudente se anima incluso a preguntar de qué va la tesis.

Por último, mi agradecimiento más especial va para Miguel, por infinidad de motivos. Ha compartido conmigo largos días de estudio; no le oí nunca quejarse cuando me iba de estancia, me venía a ver en sus vacaciones y esperaba paciente a que saliese de trabajar; en la última etapa de la tesis además me ha liberado de cualquier carga para que yo pudiese centrarme sólo en acabar; y todo esto acompañado de unos ánimos constantes y un buen humor envidiable.

Índice general

Índice de figuras

Índice de tablas

Lista de símbolos

$\odot$	Operador producto de Kronecker.
$\delta[\tau]$	Delta de Kronecker.
$\vert$	Operador concatenación.
$\otimes$	Operador entrelazado.
$*$	Operador convolución.
$\phi(n)$	Función Totient de Euler.
$\sigma(y)$	Desviación típica de una variable estadística y.
$\lfloor y \rfloor$	Redondeo del número y al entero inmediatamente inferior.
$\lceil y \rceil$	Redondeo del número y al entero inmediatamente superior.
τ	Instante de tiempo discreto.
θ	Cota máxima de correlación de un grupo de secuencias.
θ_{AC}	Cota de auto-correlación: lóbulo lateral máximo de auto-correlación de un grupo de secuencias.
θ_{CC}	Cota de correlación cruzada: valor máximo de correlación cruzada entre un grupo de secuencias.
$\boldsymbol{\Delta}$	Matriz de generación de CSS incorrelados entre sí.
$\boldsymbol{\Delta}^{(\mathbf{M},\mathbf{L_0}}$	Matriz de generación de códigos T-ZCZ compuesta por un único conjunto de M secuencias complementarias de longitud L_0.
$\boldsymbol{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0}}$	Matriz de generación de códigos T-ZCZ compuesta por M conjuntos de M secuencias complementarias de longitud L_0, incorrelados entre sí.
μ	Número de usuarios simultáneos.

π	Representación binaria de un número natural cualquiera $n = \sum_i \pi_i 2^i$. Indica el orden con el que deben concatenarse dos pares Golay incorrelados para la construcción de códigos LS según [SBH01].
π^*	Vector complementario a π, sustituye los 0 por 1 y viceversa. Se utiliza para indicar el orden con el que deben concatenarse dos pares Golay incorrelados para la construcción de códigos LS según [SBH01].
$\pi_{n,i}$	Número natural entre 0 y M que indica el orden con el que M CSS incorrelados son concatenados para la construcción de códigos LS según [ZYH05].
ϵ	Interferencia máxima permitida en la LCZ de códigos cuasi-ortogonales generalizados.
$\hat{a}$	Representación binaria $\{0,1\}$ de $a[l]$: $a[l] = (-1)^{\hat{a}[l]} \in \{-1,1\}$.
c	Velocidad de propagación de las ondas acústicas.
$C_{a,b}$	Función de correlación cruzada aperiódica entre a y b.
$D^k a$	Desplazamiento cíclico de k posiciones hacia la izquierda aplicado a la secuencia a.
DW	Ancho de bus de datos necesario para representar los valores de la señal de entrada.
E_i	Par T-ZCZ i-ésimo.
$(e_{i,0}, e_{i,1})$	Secuencias que forman el par T-ZCZ i-ésimo.
E_{AL}	Tamaño de la zona libre de interferencias en el extremo izquierdo de la SACF de códigos T-ZCZ.
E_{AR}	Tamaño de la zona libre de interferencias en el extremo derecho de la SACF de códigos T-ZCZ.
E_A	Tamaño de las zonas libres de interferencias en las terminaciones de la SACF de códigos T-ZCZ cuando $E_{AL} = E_{AR}$.
E_{CL}	Tamaño de la zona libre de interferencias en el extremo izquierdo de la SCCF de códigos T-ZCZ.
E_{CR}	Tamaño de la zona libre de interferencias en el extremo derecho de la SCCF de códigos T-ZCZ.
E_C	Tamaño de las zonas libres de interferencias en las terminaciones de la SCCF de códigos T-ZCZ cuando $E_{CL} = E_{CR}$.

f_θ	Factor de correlación no trivial, $f_\theta = 0.5 \cdot \theta_{AC} + 0.5 \cdot \theta_{CC}$.
f_L	Frecuencia de adquisición de datos en un correlador directo convencional.
f_H	Frecuencia de procesamiento de las muestras adquiridas en un correlador directo convencional.
f_{FPGA}	Frecuencia máxima de trabajo de la FPGA.
f_e	Frecuencia de emisión
f_a	Frecuencia de adquisición.
$g_k[\tau]$	Código LS k-ésimo.
$G_k(z)$	Transformada Z del código LS k-ésimo.
$\mathbf{H}$	Matriz de Hadamard.
$h_{i,j}$	Elemento j-ésimo de la fila i-ésima de una matriz de Hadamard.
$h(x)$	Polinomio primitivo usado en la generación de secuencias-m.
K	Se utiliza para indicar el número de códigos LS con zonas de correlación cero en sus funciones de correlación cruzada. Se reserva en este caso el parámetro M para indicar el número de secuencias que contienen los CSS de los cuales derivan dichos códigos LS.
L	Longitud del código bajo estudio.
l	Número natural que representa el l-ésimo bit de una secuencia.
L_0	Longitud de las secuencias complementarias cuando se utilizan para la generación de códigos ortogonales generalizados.
L_{Ms}	Longitud de una macro-secuencia de CSS.
M	Número de códigos ortogonales, incorrelados o pseudo-incorrelados de una familia. En CSS también representa el número de secuencias del conjunto.
Msc_i	Macro-secuencia obtenida mediante concatenación de las secuencias de un CSS S_i.
Mse_i	Macro-secuencia obtenida mediante entrelazado de las secuencias de un CSS S_i.
m	En CSS representa un número natural tal que $m = log_2(M)$.
m_i	En secuencias-m representa la secuencia i-ésima.

N	En CSS, número de iteraciones o etapas del algoritmo de generación. En secuencias PR, grado del polinomio primitivo de generación, esto es, número de celdas de memoria del LFSR.
N_B	Número de bits de memoria que necesita almacenar un correlador.
N_{SM}	Número de períodos del símbolo de modulación.
O_f	Factor de sobremuestreo empleado en la adquisición de una señal.
$R_{a,b}$	Función de correlación cruzada periódica entre a y b.
$r[\tau]$	Señal recibida en un correlador.
S_i	Conjunto i-ésimo de secuencias complementarias.
$s_{i,j}$	Secuencia j-ésima del conjunto i-ésimo de secuencias complementarias.
T_R	Intervalo de tiempo entre el inicio de dos emisiones codificadas.
$\overline{s_{i,j}}$	Inversión del orden de los bits de la secuencia $s_{i,j}$.
$sv_{i,j}$	Secuencia $s_{i,j}$ interpolando $M-1$ ceros entre bits consecutivos.
$W_N^{(m}$	Semilla de generación de CSS en formato binario.
$Wd_N^{(m}$	Semilla de generación de CSS en formato decimal.
$w_{i,n}$	Coeficiente i de la iteración n en las ecuaciones de generación o correlación de CSS.
W_A	Tamaño de la zona libre de interferencias alrededor del origen en la ACF de códigos ortogonales generalizados.
W_C	Tamaño de la zona libre de interferencias alrededor del origen en la CCF de códigos ortogonales generalizados.
W_q	Tamaño de la zona con interferencias reducidas alrededor del origen en códigos cuasi-ortogonales generalizados.
W	Tamaño de la zona libre de interferencias alrededor del origen en las funciones de correlación de códigos ortogonales generalizados cuando $W_A = W_C$.
$\mathbf{x}$	Posición de las ramas de salida de un ESSC.
$\overline{y}$	Media aritmética de una variable estadística y.
z^{-D_k}	Elemento de retardo.

Lista de abreviaturas

ACF *Auto-Correlation Function* (Función de auto-correlación).

ASIC *Application Specific Integrated Circuits* (Circuitos de aplicación específica).

AWGN *Additive White Gausian Noise* (Ruido aditivo blanco gaussiano.)

BPSK *Binary Phase Shift Keying* (Modulación binaria por desplazamiento de fase).

CCF *Cross Correlation Function* (Función de correlación cruzada).

CSS *Complementary Set of Sequences* (Conjunto de secuencias complementarias).

CDMA *Code Division Multiple Access* (Acceso múltiple por división en el código).

DSP *Digital Signal Processor* (Procesador de señales digitales).

DTDV Diferencia de tiempos de vuelo.

EGC *Efficient Golay Correlator* (Correlador eficiente Golay).

ETZG *Efficient Three Zero correlation zone codes Generator* (Generador eficiente de códigos T-ZCZ. Normalmente se incluye un subíndice numérico -1, 2, ó 3- que indica el método de generación y una tilde sobre el sub-índice en caso de que los códigos T-ZCZ se hayan obtenido a partir de M conjuntos complementarios incorrelados).

ETZC *Efficient Three Zero correlation zone codes Correlator* (Correlador eficiente de códigos T-ZCZ. Normalmente se incluye un subíndice numérico -1, 2, ó 3- que indica el método de generación y una tilde sobre el sub-índice en caso de que los códigos T-ZCZ se hayan obtenido a partir de M conjuntos complementarios incorrelados).

FDMA *Frequency Division Multiple Access* (Acceso múltiple por división de frecuencia).

FFT *Fast Fourier Transform* (Transformada rápida de Fourier).

FSR *Feedback Shift Register* (Registro de desplazamiento realimentado).

FPGA *Field Programmable Gate Array* (Matriz de puertas programable).

FWT *Fast Walsh Transform* (Transformada rápida Walsh).

GO *Generalized Orthogonal codes* (Códigos ortogonales generalizados).

GQO *Generalized Quasi-Orthogonal codes* (Códigos cuasi-ortogonales generalizados).

IFW *Interference Free Window* (Ventana libre de interferencias).

IW *Interference Window* (Ventana con interferencias).

ISI *Inter-Symbol Interference* (Interferencia inter-símbolo).

LA *Large Area codes* (Códigos de gran área).

LAS *Large Area Synchronized codes* (Códigos sincronizados de gran área).

LCZ *Low Correlation Zone* (Zona de bajas interferencias).

LFSR *Linear Feedback Shift Register* (Registro de desplazamiento realimentado lineal).

LS *Loosely Synchronous codes* (Códigos débilmente sincronizados).

LPS *Local Positioning System* (Sistema de posicionamiento local).

LTI *Linear Time Invariant* (Sistema lineal e invariante en el tiempo).

MAC *Multiply and Accumulation* (Operaciones de multiplicación y acumulación).

MAI *Multiple Access Interference* (Interferencia por acceso múltiple).

mcd Máximo común divisor.

M-CSS *Complementary Sets of M Sequences* (Conjuntos de M secuencias complementarias).

M-ESSG *Efficient Set of M Sequences Generator* (Generador eficiente de conjuntos de M secuencias complementarias).

M-ESSC *Efficient Set of M Sequences Correlator* (Correlador eficiente de conjuntos de M secuencias complementarias).

MF *Merit Factor* (Factor de mérito para la evaluación de las características de auto-correlación aperiódica de una secuencia).

MIMO *Multiple Input Multiple Output* (Sistemas con múltiples entradas y múltiples salidas).

OCSS *Orthogonal Complementary Sets of Sequences* (Conjuntos ortogonales de secuencias complementarias).

PPM *Pulse Position Modulation* (modulación por posición del pulso).

PR *Pseudo-Random sequences* (Secuencias pseudo-aleatorias).

PVDF Polifluoruro de Vinilideno.

QPSK *Quadrature Phase Shift Keying* (Modulación en cuadratura por desplazamiento de fase).

RF Radio Frecuencia.

SACF *Sum of Auto-Correlation Functions* (Suma de las funciones de auto-correlación).

SCCF *Sum of Cross-Correlation Functions* (Suma de las funciones de correlación cruzada).

SIC *Successive Interference Cancellation* (Cancelación por sustracción de interferencias).

SNR *Signal to Noise Ratio* (Relación señal-ruido.)

SoC *System on Chip* (Sistemas en un chip).

TDV Tiempo De Vuelo.

T-LCZ *Three Low Correlation Zone codes* (Códigos con tres zonas de reducidas interferencias).

T-ZCZ *Three Zero Correlation Zones codes* (Códigos con tres zonas de correlación cero. Con un sub-índice se indica el método -1, 2 ó 3- de construcción de dichos códigos y, cuando proceda, se añadirá una tilde sobre el sub-índice para indicar que los códigos se han obtenido a partir de M conjuntos complementarios incorrelados).

UCSS *Uncorrelated Complementary Sets of Sequences* (Conjuntos incorrelados de secuencias complementarias).

VHDL *Hardware Description Language for VHSIC* (Lenguaje de descripción Hardware para el diseño de dispositivos VHSIC).

VHSIC *Very High Speed Integrated Circuit* (Circuitos integrados de muy alta velocidad).

VLSI *Very Large Scale Integration* (Circuitos integrados a muy alta escala).

ZCZ *Zero Correlation Zone* (Zona de correlación cero).

Capítulo 1

Introducción

Existe un creciente interés en el desarrollo de sistemas sensoriales artificiales que permitan extraer información de las condiciones del entorno. Las soluciones basadas en sensores de ultrasonidos son ampliamente utilizadas debido principalmente a su bajo coste y gran precisión en condiciones ambientales estables. Sin embargo, las medidas ultrasónicas se ven afectadas negativamente por factores como cambios en la temperatura ambiente, corrientes de aire o limitaciones propias de los transductores utilizados. La búsqueda de soluciones a estos inconvenientes ha dado lugar al desarrollo de esquemas de codificación y algoritmos de proceso de la señal ultrasónica que han redundado en una mayor calidad de los resultados obtenidos.

Estos esquemas se basan en la emisión de la señal ultrasónica codificada mediante secuencias binarias con propiedades de correlación favorables, esto es, con lóbulos laterales de auto-correlación y valores de correlación cruzada próximos a cero. No obstante, no es posible eliminar completa y simultáneamente las interferencias que aparecen en dichas funciones de auto-correlación y correlación cruzada sin asignar más de una secuencia a cada emisor. Así, son muchos los trabajos dedicados a la búsqueda de las secuencias binarias más adecuadas en función de la aplicación concreta.

Si bien una correcta codificación de la señal emitida supone una mayor precisión en las medidas obtenidas, también implica el desarrollo de algoritmos de procesamiento más complejos. Esto conlleva un aumento de la carga computacional cuya implementación práctica puede llegar a superar los límites impuestos por la necesidad de trabajar en tiempo real. Por tanto, el esquema de codificación y sus algoritmos de proceso asociados deben ser elegidos de modo que reduzcan el número de operaciones a llevar cabo, y sus tiempos de ejecución cumplan las restricciones derivadas del trabajo en tiempo real.

En esta tesis se determina el esquema de codificación más adecuado para un sistema sensorial basado en la medida de tiempos de vuelo de señales ultrasónicas. Para ello se analizan las propiedades de correlación aperiódica de distintos tipos de códigos binarios, incluyendo los códigos ortogonales generalizados de más reciente aparición. Asimismo, se propone un nuevo esquema de codificación con lóbulos laterales nulos en determinadas zonas de sus funciones de correlación aperiódicas. Finalmente, se desarrolla toda la algoritmia asociada a la generación y detección eficiente de los códigos propuestos y de aquellos derivados de conjuntos de secuencias complementarias (CSS, *Complementary Set of Sequences*).

Cabe destacar que, aunque los algoritmos de correlación eficiente propuestos han sido desarrollados para mejorar las prestaciones de sistemas sensoriales ultrasónicos, su uso puede extenderse a otros campos tales como la telefonía móvil y comunicaciones inalámbricas, en donde mediante técnicas de Acceso Múltiple por División en el Código (CDMA) se evitan interferencias entre usuarios.

1.1. Entorno de desarrollo de la tesis

Este trabajo de tesis se ha llevado a cabo en el Departamento de Electrónica de la Universidad de Alcalá, que acumula una amplia experiencia en el diseño de sistemas sensoriales aplicados a robótica móvil. En concreto, la tesis se ha valido de la experiencia adquirida en varios proyectos de investigación desarrollados en dicho Departamento:

- PARMEI (*Posicionamiento absoluto de robots móviles en espacios interiores*; referencia DIP2003-08715-C02-01), este proyecto se desarrolló conjuntamente con el Instituto de Automática Industrial del Consejo Superior de Investigaciones Científicas (CSIC) durante los años 2004-2006. El objetivo principal de este proyecto fue la elaboración de nuevos métodos y técnicas para determinar la posición absoluta de robots móviles en espacios interiores; para ello se utilizaron sensores de ultrasonidos e infrarrojos distribuidos en el entorno de movimiento del robot. Los códigos empleados fueron pseudo-aleatorios tipo Gold y conjuntos de secuencias complementarias. En ambos casos se consiguió separar la información proveniente de cada emisor mediante CDMA, así como trabajar en condiciones de baja potencia emitida y eliminar los problemas multicamino propios del empleo de señales ultrasónicas en espacios interiores.

- INCUBUS (*Estructuras sensoriales ultrasónicas de alto ancho de banda para aire: diseño y proceso de información*; referencia: CAM-UAH2006/016). Las tareas asociadas a este proyecto fueron realizadas durante el año 2007, con el propósito de desarrollar nuevas técnicas de codificación y procesamiento de señales ultrasónicas para el mejor aprovechamiento de la información contenida en los parámetros espectrales de la señal, de cara a la localización y clasificación de reflectores.

- RESELAI (*Integración de redes de sensores acústicos, de visión y RFID para localización en ambientes inteligentes*; referencia: TIN2006-14986-CO2-01)), se trata de un proyecto multidisciplinar en el que colabora también el Instituto de Automática Industrial del CSIC. El proyecto comenzó en octubre del año 2006 y su finalización está prevista para octubre de 2009. Los esfuerzos del proyecto se orientan a aportar funciones de detección de presencia, localización física e interacción entre usuarios; para ello se cuenta con tres tecnologías sensoriales complementarias: acústica, cámaras de visión y RFID. Se abordan los aspectos sensoriales, de procesamiento de señal, y algoritmos avanzados de alto nivel; así como aspectos de escalabilidad, coste de integración y mantenimiento, comunicación de datos y flexibilidad de las redes sensoriales diseñadas.

- HERSO (*Estudio de modelos de propagación y técnicas MIMO en radio*), otros participantes en este proyecto son las empresas INDRA Sistemas S. A. y GCM Communications Technology. Su duración va desde noviembre de 2007 a mayo de 2009. Sus objetivos incluyen el estudio de modelos MIMO de propagación identificados a partir del empleo de CSS y códigos derivados (LS y T-ZCZ).

Finalmente, desde el punto de vista económico, esta tesis ha sido posible gracias a la financiación de la Comunidad de Madrid a través de una beca de formación de personal investigador (*FPI-CAM*) adjudicada en la convocatoria de 2005 y ayudas para la realización de estancias breves en el extranjero. Las estancias subvencionadas, de gran utilidad para el desarrollo de la tesis, han tenido lugar durante los años 2006 y 2007 en el Departamento de Ingeniería Eléctrica y Electrónica de la Universidad College Cork (Irlanda), y en el Laboratorio de Percepción Activa de la Universidad de Amberes (Bélgica).

1.2. Estructura de la tesis

La redacción de esta tesis se ha dividido en siete capítulos atendiendo a los diferentes temas que han sido abordados en el desarrollo de la misma.

Después de esta introducción, en el capítulo 2 se lleva a cabo una revisión detallada de los códigos binarios de mayor relevancia en aplicaciones de codificación, comprobando además el interés en la temática a partir del número de publicaciones asociadas. A continuación, se describen los trabajos más destacados relacionados con sistemas ultrasónicos que codifican la señal emitida para obtener una mejora en su comportamiento. Por otro lado, se revisan las opciones de detección de los códigos anteriores mediante métodos eficientes que supongan una reducción del número de operaciones a realizar. Posteriormente, se hace un breve repaso de las arquitecturas de computación candidatas a implementar el procesamiento eficiente de los códigos binarios cuando se usan en aplicaciones ultrasónicas. Una vez constatado que los sistemas configurables son los más adecuados, se presentan distintos trabajos que hacen uso de estos sistemas para el tratamiento de la señal ultrasónica. Finalmente, teniendo en cuenta las limitaciones de los códigos binarios actuales, se ha procedido a establecer los objetivos y aportaciones de esta tesis.

En el capítulo 3 se lleva a cabo un análisis comparativo de los códigos binarios que mejor pueden adaptarse a un sistema ultrasónico con múltiples emisiones aperiódicas y detección asíncrona o cuasi-síncrona. Así, se han evaluado las propiedades de correlación de códigos Kasami, CSS, códigos LS y códigos con tres zonas de correlación cero T-ZCZ. Para todos ellos se proporciona un listado con las secuencias específicas cuya emisión simultánea redunda en una menor interferencia por acceso múltiple.

El capítulo 4 acomete la implementación de los algoritmos de detección y generación de CSS y códigos LS sobre una plataforma configurable. Para el primero de los casos, se propone una modificación de los algoritmos eficientes ya existentes que permite una implementación hardware genérica. Por otro lado, y teniendo en cuenta las restricciones en la emisión simultánea de todas las secuencias complementarias, se han propuesto esquemas de emisión mediante concatenación o entrelazado, adaptando el correlador implementado a dichas estrategias de emisión. En cuanto a los códigos LS, se desarrollan en este capítulo nuevos algoritmos de generación y correlación para que el tratamiento de los mismos pueda efectuarse en el menor tiempo posible, respetando de este modo las restricciones temporales propias de cualquier sistema de procesamiento ultrasónico en tiempo real. Las mejoras derivadas del uso de los algoritmos e implementaciones propuestas se han verificado mediante comparación con una implementación directa de los correladores.

En el capítulo 5 se aborda un nuevo esquema de codificación con parejas de códigos cuya suma de funciones de correlación presenta tres áreas sin interferencias en caso de emisión aperiódica, y cuya emisión periódica no requiere de zonas de guarda para mantener la ortogonalidad alrededor del origen. Se proponen varios métodos de generación de dichas parejas, y para todos ellos se desarrollan algoritmos de detección eficiente que reducen sobremanera el número de operaciones a realizar frente a las requeridas por un correlador directo. Además, las arquitecturas propuestas pueden ser fácilmente implementadas en hardware configurable.

El capítulo 6 presenta el diseño de un sistema de posicionamiento local (LPS) ultrasónico basado en los esquemas de codificación evaluados en la tesis (códigos Kasami, CSS, LS y T-ZCZ). La posición de un robot móvil se calcula a partir de las diferencias de tiempos de vuelo de las señales codificadas emitidas por un conjunto de cinco balizas situadas en el techo. El diseño del LPS incluye la modulación de los códigos emitidos para adecuarlos a las características de los transductores empleados. Además se presenta el diseño e implementación de las etapas que conforman el módulo de detección. Una primera etapa, en la que se encuentran integrados los correladores descritos en capítulos anteriores, está dedicada a la medida de tiempos de vuelo de las recepciones procedentes de las balizas; y una segunda etapa es la encargada de realizar las tareas de posicionamiento de más alto nivel. Las prestaciones de los diferentes códigos y los niveles de ruido soportados por el sistema se han comprobado mediante un conjunto de simulaciones, para finalmente, verificar los resultados obtenidos mediante pruebas reales.

Por último, en el capítulo 7 se resumen las conclusiones finales y aportaciones realizadas en esta tesis. Asimismo, se describen posibles líneas futuras de investigación que pueden derivarse o aparecer a partir del trabajo realizado. Se incluye además un listado de las publicaciones consecuencia del trabajo de investigación llevado a cabo.

Capítulo 2

Estado de la cuestión y objetivos planteados

La mayoría de los sistemas ultrasónicos basados en la medida de tiempos de vuelo codifican la emisión mediante secuencias con propiedades similares a las del ruido blanco gaussiano, que son luego detectadas en la señal recibida mediante técnicas de correlación. Con esto se consigue mejorar, entre otros aspectos, la precisión temporal, la resolución espacial y la relación señal-ruido. Por otro lado, si las secuencias son elegidas de modo que el valor de correlación cruzada entre ellas sea mínimo, será posible realizar varias emisiones simultáneas sin que se produzcan interferencias entre ellas. Esto, además de permitir un aumento de la frecuencia de operación, resulta de gran utilidad en sistemas que requieren conjuntos de medidas obtenidas en condiciones similares y en donde dichas condiciones cambian rápidamente. Éste es el caso, por ejemplo, de los sistemas de localización, en donde el robot u objeto a localizar suele estar en movimiento. Es importante también disponer de algoritmos de correlación eficientes que, además de reducir el número de operaciones a realizar, puedan implementarse fácilmente en sistemas configurables, permitiendo así el procesamiento en tiempo real de las señales recibidas.

La primera sección de este capítulo presenta las ventajas asociadas a la codificación de la señal ultrasónica. A continuación, en la segunda sección, se realiza un resumen de los códigos binarios más empleados. Seguidamente se presenta una revisión de los trabajos más relevantes que emplean emisiones ultrasónicas codificadas, y un breve apunte acerca de otras posibles aplicaciones de los códigos binarios. A continuación en la quinta sección se comentan los esquemas de correlación eficiente existentes asociados a dichos códigos, y se justifica la elección de arquitecturas configurables como plataformas de implementación de los algoritmos de tratamiento de la señal ultrasónica. Finalmente, se evalúa el esfuerzo investigador invertido en el estudio y mejora de técnicas de codificación, y se presenta una descripción detallada de los objetivos planteados en la tesis.

2.1. Codificación de la señal ultrasónica

Los sistemas sensoriales ultrasónicos destacan frente a otras alternativas, tales como las basadas en radio o visión artificial, debido a que permiten realizar medidas de distancias de gran precisión usando sensores de bajo coste. La figura 2.1 resume la precisión alcanzada con las tecnologías sensoriales actuales en aplicaciones de localización.

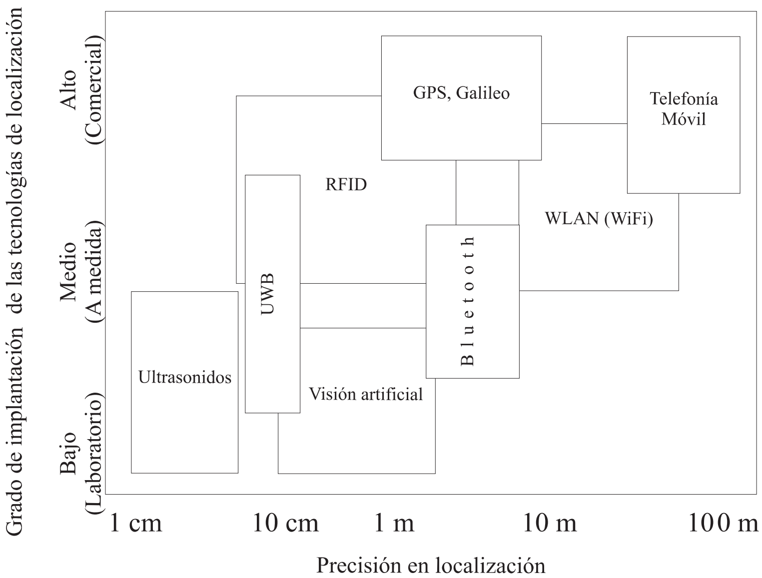


Figura 2.1: Clasificación de las técnicas de localización atendiendo a la precisión alcanzable y su grado de implantación. La altura de las cajas representa la evolución que se estima para cada tecnología en los próximos años [HSK04].

En general, estos sistemas sensoriales basados en señales ultrasónicas obtienen la medida de distancias a partir del tiempo de vuelo (TDV) de dichas señales. Esta técnica consiste en medir el tiempo que transcurre desde que la señal es emitida hasta que es recibida o bien hasta que se refleja en un objeto del entorno, según se trate de una aplicación de localización o de detección de obstáculos, respectivamente. Ahora bien, las señales ultrasónicas en el aire son sensibles a factores de diversa naturaleza que influyen negativamente en la precisión de las medidas de TDV. En [BK91a][Eve95] se enumeran algunos de estos factores:

- *Variaciones en la velocidad de propagación de los ultrasonidos.* Para el cálculo de distancias a partir del TDV se necesita conocer la velocidad de propagación c de los ultrasonidos en el medio; de este modo, la distancia r entre un emisor y un receptor se calcula según la expresión $r = c \cdot TDV$. La velocidad del sonido en el aire depende de la humedad y, fundamentalmente, de la temperatura ambiente, lo que puede provocar errores en la determinación de los TDV. Deben emplearse, por tanto, técnicas de compensación, como la medida directa de la temperatura ambiente y la posterior aplicación de fórmulas de corrección. Además, en ambientes externos debe considerarse el efecto del viento, que es especialmente crítico en aplicaciones en las que se requieren varias medidas realizadas bajo condiciones similares. En el caso de velocidades de viento no muy elevadas, la velocidad aparente del sonido se calcula sumando la velocidad de propagación y la componente del viento en la dirección de propagación. En caso de velocidades elevadas, debe considerarse además la componente normal del viento.

- *Atenuación del ultrasonido en la atmósfera.* La divergencia geométrica, la absorción atmosférica, la presencia de niebla o lluvia, la refracción asociada a los gradientes de temperatura y viento y las turbulencias pueden provocar la atenuación de la onda ultrasónica. Para velocidades de viento menores que 10 m/s son los dos primeros mecanismos los que dominan, pudiéndose despreciar el resto a efectos prácticos. Para velocidades mayores, es la atenuación debida a la refracción la que puede llegar a ser considerable [Á05]. Debido a esta atenuación, las técnicas basadas en el análisis de la forma de onda recibida, como la propuesta en [KK95], no resultan adecuadas.

- *Inexactitudes en los sistemas de medida y temporización.* La implementación de los temporizadores que calculan los TDV se lleva a cabo a partir de contadores con un reloj de frecuencia determinada. Este reloj impone la exactitud máxima con la que pueden realizarse las medidas de tiempo.

- *Interacción con las superficies.* Cuando el ultrasonido incide sobre una superficie parte de su energía es reflejada, la cantidad reflejada y su dirección dependen del ángulo de incidencia de la onda ultrasónica y de la naturaleza de la superficie. En entornos cerrados predominan los objetos manufacturados por el hombre (paredes, sillas, estanterías, etc.) de superficie lisa, y por tanto de reflexión especular [KK95], lo que puede provocar la recepción de ecos tras múltiples rebotes, en lugar del eco directo del objeto más cercano. En entornos exteriores, sin embargo, se encuentran en mayor número las superficies rugosas. En [BK91b] se modela este tipo de superficies para el caso de usar transductores ultrasónicos de banda ancha.

Los inconvenientes citados anteriormente han sido ampliamente estudiados y documentados, habiéndose propuesto modelos que permiten la evaluación y, en algunos casos, compensación de los mismos. Como ejemplo, en [KK95] se usa un modelo lineal para caracterizar el efecto de los transductores, la excitación del sistema y los fenómenos de absorción y dispersión, con objeto de mejorar las prestaciones de un sistema sónar de clasificación y localización de planos, esquinas y salientes en espacios interiores. En [Á05], sin embargo, se estudian los fenómenos que afectan a la propagación de los ultrasonidos en ambientes exteriores. En este mismo trabajo se propone la codificación de la señal emitida con conjuntos de cuatro secuencias complementarias para mejorar el desempeño de estos sistemas.

El empleo de técnicas de compresión de pulsos y filtrado óptimo parecidas a las usadas en radar es un método habitual para aumentar la precisión de las medidas en un sistema ultrasónico. Estas técnicas consisten en la emisión de una señal codificada, cuyo eco es buscado y localizado en la señal recibida mediante correlación [HG88]. Se consigue así detectar el eco recibido aun en condiciones de ruido elevado. Basta con incrementar la longitud de los códigos de codificación, y no su amplitud, para incrementar la relación señal-ruido a la salida del detector. Por otro lado, aumenta notablemente la precisión temporal y la resolución espacial frente a métodos de cálculo de TDV mediante técnicas de integración y umbralización (véase [Pol91]). Esto es debido a que el instante de llegada del eco viene determinado por un único pulso y no por una envolvente, y a que la correlación de dos ecos solapados genera dos pulsos distintos y fáciles de discriminar.

Además, si las señales escogidas para la codificación presentan valores bajos de correlación cruzada, es posible la emisión simultánea sin que se produzcan interferencias o *crosstalk* entre ellas. Esta característica resulta de gran importancia en el caso de disponer de varios transductores emitiendo al mismo tiempo y sobre la misma zona espacial. De este modo se obtienen simultáneamente medidas desde distintas posiciones y, por tanto, una mayor información del entorno. Asimismo, con las técnicas de compresión de pulsos, es posible la emisión de una señal sin necesidad de esperar a que sean recibidos todos los ecos emitidos anteriormente, aumentando por tanto la frecuencia de operación proporcionalmente al número de señales incorreladas o pseudo-incorreladas disponibles. Del mismo modo, en [Her03] se demuestra que, dependiendo de las propiedades de auto-correlación de las señales emitidas, puede reducirse casi en su totalidad el límite inferior para la distancia mínima que puede medirse con un único transductor.

2.2. Códigos binarios empleados en la codificación

Antes de revisar los códigos de compresión de pulsos que más aceptación han tenido en el campo del sónar, así como otros de reciente aparición en el sector de las comunicaciones, se repasarán algunos conceptos a los que se hará referencia a lo largo de la tesis. En primer lugar es importante señalar que la nomenclatura utilizada en esta tesis es la introducida por Pingzhi Fan en [FD96], cuyo uso se está extendiendo entre los autores que trabajan con códigos con zonas de correlación cero [SYKA98, FH00, Fan04, RC04]. La ventaja de esta nomenclatura es que permite distinguir, además del grado de correlación entre códigos, el grado de sincronismo necesario para la correcta detección de los mismos.

Se calcula la correlación entre dos señales con el objeto de medir su grado de similitud. Esto es, la correlación es una medida de la dependencia de una señal con otra o consigo misma. Dada una familia A con M códigos binarios unitarios[1] de longitud L $\{A = a_m[l] \in \{-1, 1\};\ 0 \leq m \leq M-1;\ 0 \leq l \leq L-1\}$, la función de auto-correlación (ACF) discreta periódica se obtiene de la expresión (2.1) cuando $m = s$ y la función de correlación cruzada

[1] Códigos *unitarios* son aquellos que asignan una única secuencia a cada emisor [CLY+06].

(CCF) periódica cuando $m \neq s$:

$$R_{a_m,a_s}[\tau] = \sum_{l=0}^{L-1} a_m[l] a_s[l+\tau] \tag{2.1}$$

En la expresión anterior las secuencias a_m y a_s son periódicas, es decir, $a_m = (\cdots, a_m[0], a_m[1], \cdots, a_m[L-1], a_m[0], a_m[1], \cdots, a_m[L-1], \cdots)$, por lo que la suma $l+\tau$ se realiza en *módulo L*. La función de correlación aperiódica viene definida, sin embargo, por (2.2).

$$C_{a_m,a_s}[\tau] = \begin{cases} \sum_{l=0}^{L-1-\tau} a_m[l] a_s[l+\tau], & 0 \leq \tau \leq L-1 \\ \sum_{l=0}^{L-1+\tau} a_m[l-\tau] a_s[l], & 1-L \leq \tau < 0 \\ 0, & |\tau| \geq L \end{cases} \tag{2.2}$$

Cuando $m = s$ se obtiene la ACF aperiódica, y cuando $m \neq s$ la CCF aperiódica. Normalmente, y con objeto de simplificar, se considera la función de correlación aperiódica sólo en el rango $0 \leq \tau \leq L-1$.

Para el desfase cero el valor de las funciones de correlación periódica y aperiódica coinciden ($C_{a_m,a_s}(0) = R_{a_m,a_s}(0)$ y $C_{a_m,a_m}(0) = R_{a_m,a_m}(0) = \sum_{l=0}^{L-1} a_m[l]^2$); es decir, el valor del pico principal de auto-correlación es el mismo se trate o no de emisiones periódicas.

Un criterio ampliamente usado para evaluar las bondades de una familia de códigos es calcular su *cota máxima de correlación* (en inglés *bound*):

$$\theta = \max\{\theta_{AC}, \theta_{CC}\} \tag{2.3}$$

En donde θ_{AC} representa el valor del máximo pico lateral obtenido de entre todas las auto-correlaciones de los M códigos de la familia, y θ_{CC} es el pico de correlación cruzada mayor entre ellas:

$$\theta_{AC} = \max\left\{\frac{|R_{a_m,a_m}[\tau]|}{R_{a_m,a_m}[0]}; \forall m \in [0, \cdots, M-1]; \forall \tau \neq 0\right\} \tag{2.4}$$

$$\theta_{CC} = \max\left\{\frac{|R_{a_m,a_s}[\tau]|}{R_{a_m,a_m}[0]}; \forall m, s \in [0, \cdots, M-1]; m \neq s; \forall \tau\right\} \tag{2.5}$$

Sustituyendo en las ecuaciones (2.4) y (2.5) R_{a_m,a_m} y R_{a_m,a_s} por C_{a_m,a_m} y C_{a_m,a_s} respectivamente se obtienen las cotas para el caso de correlaciones aperiódicas.

Los códigos ideales tendrían un pico muy elevado en la ACF para $\tau = 0$ y lóbulos laterales nulos en $\tau \neq 0$, según se indica en (2.6). Además estarían incorrelados, esto es, su CCF sería cero para todos los posibles desplazamientos, eliminando de este modo cualquier interferencia entre ellos.

$$R_{a_m,a_s}[\tau] = \begin{cases} L, & \tau = 0, & m = s \\ 0, & 1 \leq \tau \leq L-1, & m = s \\ 0, & 0 \leq \tau \leq L-1, & m \neq s \end{cases} \tag{2.6}$$

No obstante, es imposible encontrar familias de códigos unitarios con ACF y CCF ideales simultáneamente [Wel74, FD96]. Como se verá más adelante, los conjuntos complementarios de secuencias y los pares Golay logran este comportamiento ideal a partir de la suma de las correlaciones de un determinado número de secuencias. En estos casos cada emisor tiene asignadas más de una secuencia, utilizándose esquemas de modulación o reordenación de bits que permiten, en la medida de lo posible, la transmisión simultánea de las mismas. Estos esquemas, sin embargo, implican la aparición de interferencias en las ACF y CCF [DMUH+06]. Las interferencias debidas a los valores no nulos de los lóbulos laterales de auto-correlación se conocen como interferencias inter-símbolo (ISI); y el hecho de que las correlaciones cruzadas entre códigos no sean cero provoca interferencias por acceso múltiple (MAI). Para disminuir las ISI y MAI deben elegirse códigos con cotas de correlación mínimas.

Se dice que dos códigos son ortogonales si cumplen:

$$R_{a_m,a_s}[\tau] = \begin{cases} L, & \tau = 0, \quad m = s \\ 0, & \tau = 0, \quad m \neq s \end{cases} \tag{2.7}$$

Nótese que la expresión anterior asegura que la CCF es nula únicamente para el desplazamiento cero[2]. Éste es el caso de los códigos Walsh, que tienen valores muy altos en las ACF y CCF en caso de pérdida de sincronismo (cuando $\tau \neq 0$) [Woo02]. Además de la dificultad de mantener sincronizados emisores y receptores, posibles reflexiones provocarían la recepción de la señal emitida en distintos instantes de tiempo destruyendo la ortogonalidad entre los códigos. Esta situación, en la que la señal emitida puede propagarse a través de distintos caminos (debido, por ejemplo a reflexiones en el suelo o en obstáculos) se conoce como *propagación multicamino*.

Como solución, Pingzhi Fan [FH00, Fan04] acuña dos nuevos conceptos: *ortogonalidad generalizada* o *Z-ortogonalidad* y *cuasi-ortogonalidad generalizada*, que pueden ser empleados para reducir las ISI y MAI en entornos que no requieren un sincronismo estricto (también denotados como entornos cuasi-síncronos).

Dos códigos son ortogonales generalizados (GO, *Generalized Orthogonal*) o Z-ortogonales si tienen las siguientes características:

$$R_{a_m,a_s}[\tau] = \begin{cases} \eta L, & \tau = 0, & m = s \\ 0, & 1 \leq |\tau| \leq W, & m = s \\ 0, & 0 \leq |\tau| \leq W, & m \neq s \end{cases} \tag{2.8}$$

$$\eta = \frac{1}{L} \sum_{l=0}^{L-1} |a_m[l]| \leq 1 \tag{2.9}$$

[2]Otros autores [TL72, ÁUM+06] hablan de códigos ortogonales cuando su CCF es nula para cualquier desplazamiento. En este trabajo se dirá que dichos códigos se encuentran incorrelados.

En donde W representa una zona alrededor del origen donde las funciones de correlación tienen valor cero. Huelga decir que cuanto mayor sea W mejores características presentarán los códigos. Por el contrario, si $W = 0$ se obtienen códigos ortogonales convencionales. En definitiva, W indica el grado de ortogonalidad generalizada de los códigos, y es conocida como *zona de correlación cero* (ZCZ, *Zero Correlation Zones*) [FSKD99] o *ventana libre de interferencias* (IFW, *Interference Free Window*) [Li03]. Más concretamente, la IFW hace alusión al área de tamaño $2W + 1$ alrededor del origen. El parámetro η es uno cuando $a_m \in \{-1, 1\}$; aunque si hay valores nulos en el código, esto es si $a_m \in \{-1, 0, 1\}$, será menor que la unidad.

En el caso de códigos cuasi-ortogonales generalizados (GQO, *Generalized Quasi-Orthogonal*) las condiciones planteadas en (2.8) se relajan, de modo que las interferencias alrededor del origen no son cero, sino que su valor está por debajo de un umbral ϵ aceptable por el sistema. Con esto se consigue aumentar el número de códigos que cumplen esta condición, a costa de asumir un conjunto de interferencias controladas.

$$R_{a_m,a_s}[\tau] = \begin{cases} \eta L, & \tau = 0, \quad m = s \\ \leq \epsilon, & 1 \leq |\tau| \leq W_q, \quad m = s \\ \leq \epsilon, & 0 \leq |\tau| \leq W_q, \quad m \neq s \end{cases} \tag{2.10}$$

En donde W_q se denota como *zona de reducidas interferencias* (LCZ, *Low Correlation Zone*) y η se define según (2.9).

Por simplicidad, las expresiones (2.6) a (2.10) se han planteado sólo para el caso de emisiones periódicas, siendo similares las definiciones del caso aperiódico.

2.2.1. Códigos Walsh

Ampliamente utilizados en sistemas de comunicación muti-usuario síncronos, los códigos Walsh [Wal23] son binarios ortogonales y pueden generarse fácilmente a partir de matrices de Hadamard (se habla entonces de códigos Walsh-Hadamard). Una matriz de Hadamard $\mathbf{H}$, con elementos $h_{i,j} \in \{-1, 1\}$, es una una matriz cuadrada binaria ortogonal por columnas, esto es, el producto escalar de dos columnas cualesquiera es nulo. La primera fila de estas matrices está constituida por 1, y el resto tienen el mismo número de -1 que de 1. Pueden obtenerse recursivamente según el algoritmo de J. J. Sylvester [Syl67]:

$$H_1 = (1) \qquad H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad H_L = \begin{pmatrix} H_{L/2} & H_{L/2} \\ H_{L/2} & -H_{L/2} \end{pmatrix} \tag{2.11}$$

Cada fila de una matriz de Hadamard $L \times L$ es un código Walsh-Hadamard, de modo que el conjunto de códigos Walsh-Hadamard ortogonales está formado por M códigos de longitud $L = M$, siendo L cualquier potencia de dos.

Según se ha comentado anteriormente, estos códigos tienen aplicación sólo en el caso que haya un perfecto sincronismo entre emisores y receptores. Si se recibe el código con un desfase mínimo de un bit aparecen interferencias que pueden ser de la misma magnitud que el pico de correlación (algunas secuencias son sólo versiones desfasadas de otras). Mantener un sincronismo tan preciso es complicado ya que, debido al multicamino, las señales llegan a los receptores con desviaciones de tiempo aleatorias.

Por otro lado, los códigos Walsh-Hadamard (y en general todos los códigos ortogonales) presentan valores de correlación bastante pobres para desfases distintos de cero, según se muestra en la figura 2.2. Esto, unido a la dificultad de una perfecta sincronización, conducen al agravamiento del efecto *cerca-lejos*. Este efecto ocurre cuando la energía de la señal emitida por un emisor aumenta considerablemente en el receptor debido a su proximidad a éste. Como consecuencia, señales más débiles procedentes de emisores alejados podrían no ser detectadas, ya que sus picos de auto-correlación quedarían enmascarados por los picos de correlación-cruzada de la señal recibida con mayor energía.

Puede observarse además en la figura 2.2 que el máximo de correlación se encuentra retardado L muestras, esta variación del resultado teórico se corresponde con el funcionamiento real del sistema en donde se obtiene el pico de correlación una vez que se han procesado las secuencias. Se asumirá este retardo L en las siguientes representaciones.

2.2.2. Códigos Barker

Los códigos Barker [Bar53] se definen como una secuencia de dígitos binarios $\{B = b[l] \in \{-1, 1\};\ 0 \leq l \leq L-1;\ L \geq 2\}$, que cumplen la siguiente propiedad:

$$|C_{B,B}[\tau]| = \left| \sum_{l=0}^{L-1-\tau} b[l]b[l+\tau] \right| \leq 1 \qquad 1 \leq \tau \leq L-1 \tag{2.12}$$

Estos códigos presentan dos inconvenientes importantes. Por un lado, sólo resulta posible generar códigos de longitudes $L = 2,\ 3,\ 4,\ 5,\ 7,\ 11$ y 13 (véase la tabla 2.1), por lo que el pico máximo de auto-correlación que puede conseguirse con estas secuencias es igual a 13. Por otro lado, para cada longitud existe un único código (a excepción de las longitudes 2 y 4 que disponen de dos), así que no es posible encontrar códigos con correlaciones cruzadas bajas entre ellos.

En la figura 2.3 se muestra la ACF de un código Barker de $L = 13$ bits.

Se han hecho numerosos esfuerzos en la búsqueda de códigos Barker de mayor longitud; así en [CG96] se propone un conjunto de secuencias Barker polifásicas (hasta 7200 fases) con longitudes $L \leq 31$. No obstante esta longitud sigue siendo insuficiente en muchas aplicaciones y, lo que es peor, el coste hardware asociado a un correlador complejo de estas características es muy alto.

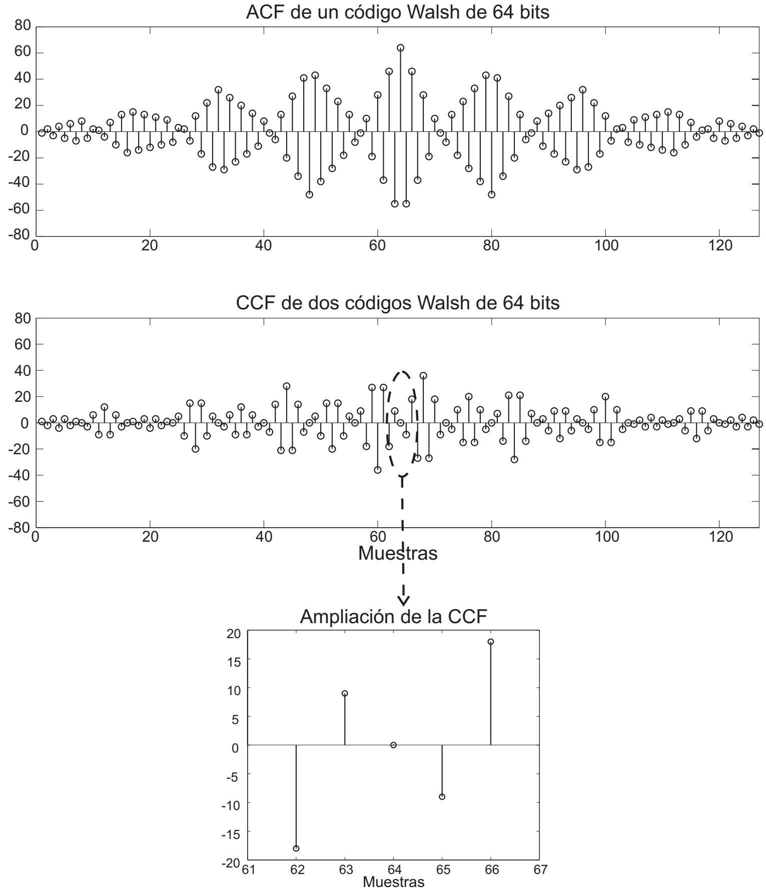


Figura 2.2: ACF y CCF de códigos Walsh de 64 bits.

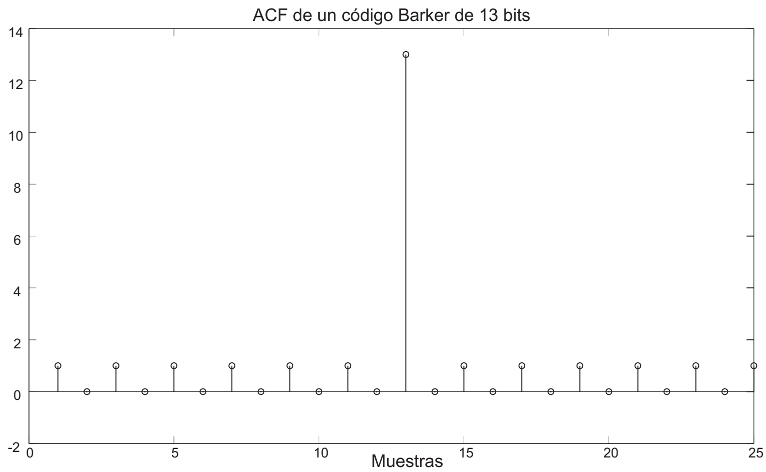


Figura 2.3: Auto-correlación de un código Barker de 13 bits.

Símbolo	Longitud	Códigos Barker
B_2	2	[1 -1], [1 1]
B_3	3	[1 1 -1]
B_4	4	[1 1 -1 1], [1 1 1 -1]
B_5	5	[1 1 1 -1 1]
B_7	7	[1 1 1 -1 -1 1 -1]
B_{11}	11	[1 1 1 -1 -1 -1 1 -1 -1 1 -1]
B_{13}	13	[1 1 1 1 1 -1 -1 1 1 -1 1 -1 1]

Tabla 2.1: Conjuntos de códigos Barker.

En [CL95] se plantean códigos Barker GQO para aplicaciones cuasi-síncronas. Se consigue con esto un mayor número de códigos de la misma longitud y longitudes mayores a 13, a costa de asumir interferencias altas fuera de la LCZ -véase la expresión (2.10)-. Cuando $\epsilon = 1$ y $W_q = L - 1$ se obtienen los códigos Barker originales. Para que estos códigos sean de utilidad debe cumplirse $2L \geq W_q^2 + 3\epsilon(\epsilon - 1) + 1$, lo que implica usar valores bajos de W_q y ϵ para minimizar las MAI. Sin embargo, reducir ϵ conlleva disponer de un menor número de códigos de igual longitud, disminuyendo la capacidad multi-usuario del sistema. Aumentar ϵ y reducir W_q podría ser una solución, aunque debe entonces aumentarse la precisión del sincronismo entre códigos ya que no pueden recibirse fuera de la LCZ.

Un método muy sencillo para incrementar la longitud de los códigos Barker consiste en realizar el producto de Kronecker de dos de ellas (por ejemplo: $B_{45} = B_4 \odot B_5$=[1 1 1 -1 1 1 1 1 -1 1 -1 -1 -1 1 -1 1 1 1 -1 1]) [Mah00]. A estos códigos se les ha denominado como *Barker extendidos* y su única ventaja radica en su facilidad de generación. A cambio, los lóbulos laterales en la ACF son ahora mayores que la unidad y continúa sin haber códigos con baja correlación cruzada entre ellos.

2.2.3. Secuencias pseudo-aleatorias

Bajo este nombre se engloban todas las secuencias binarias que presentan un comportamiento similar al de una secuencia completamente aleatoria, aunque son generadas de forma determinista. Para conocer el grado de aleatoriedad de las secuencias se definen los siguientes criterios:

- La ACF periódica en cualquier instante $\tau \neq 0$ debe ser una constante c de bajo valor.

$$R_{A,A}[\tau] = \begin{cases} L, & \tau = 0 \\ c, & \tau \neq 0 \end{cases} \tag{2.13}$$

En donde $\{A = a[l] \in \{-1, 1\}; 0 \leq l \leq L - 1\}$.

- En cada período el número de 1 no debe exceder en más de una unidad al número de -1, esto es, $|\sum_{l=0}^{L-1} a[l]| \leq 1$.

- En cada período la mitad de los 1 ó -1 consecutivos tienen longitud 1, un cuarto tienen longitud 2, un octavo longitud 3, etc. Además, hay el mismo número de 1 consecutivos que de -1.

Sólo unas pocas secuencias cumplen estrictamente estas propiedades, por lo que se permite una cierta flexibilidad a la hora de aplicar estos criterios. De este modo, secuencias binarias cuya ACF presenta más de dos valores también son consideradas pseudo-aleatorias (PR, *Pseudo-Random sequences*).

Muchas de estas secuencias se generan gracias a una estructura denominada registro de desplazamiento realimentado (FSR, *Feedback Shift Register*), cuya implementación en hardware digital es relativamente sencilla. Un FSR está constituido por dos bloques: un registro de desplazamiento y una función de realimentación que genera el bit de mayor peso. A cada impulso de una señal de reloj todos los bits del registro se desplazan una posición a la derecha y el nuevo bit de mayor peso se genera a partir del valor de los bits del registro mediante la función de realimentación. La función de realimentación puede ser tan sencilla como realizar la operación XOR con bits específicos del registro de desplazamiento; en este caso se dice que el FSR es lineal (LFSR). La figura 2.4 representa un sistema de este tipo para el caso de usar un registro de desplazamiento con N celdas de memoria. La función de realimentación o *polinomio primitivo* $h(x)$ de grado N se representa como $h(x) = h[0] \cdot x^N + h[1] \cdot x^{N-1} + \cdots + h[N-1] \cdot x + h[N] = \sum_{i=0}^{N} h[i] \cdot x^{N-i}$, siendo $h[0] = h[N] = 1$. Un valor $h[k] = 1$ indica que existe realimentación a la salida de la k-ésima celda, y un valor $h[k] = 0$ significa lo contrario. Por otro lado, si el estado inicial del LFSR es $\hat{a}_0[N-1], \hat{a}_0[N-2], \cdots, \hat{a}_0[0]$, entonces la secuencia $\hat{a}[n]$ generada puede representarse de forma recursiva según (2.14).

$$\hat{a}[n] = \begin{cases} \hat{a}_0[n], & 0 \leq n < N \\ \sum_{i=1}^{N} h[i] \cdot \hat{a}[n-i], & n \geq N, \ h[i] \in 0, 1 \end{cases} \tag{2.14}$$

Siendo $\hat{a}[n] \in \{1, 0\}$ la representación binaria de la secuencia $a[n] = (-1)^{\hat{a}[n]} \in \{-1, +1\}$.

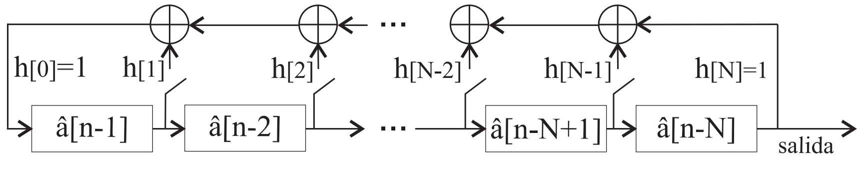


Figura 2.4: Registro de desplazamiento realimentado lineal (LFSR).

Secuencias-m

Las secuencias generadas con un LFSR de N bits pueden alcanzar un período máximo de $2^N - 1$ bits sin que se repita el patrón. La secuencia todo-ceros debe evitarse ya que es absorbente (todas las celdas de memoria del registro de desplazamiento estarían a cero y no sería posible salir de ese estado). Las secuencias que tienen esta longitud máxima se denominan *secuencias−m* [Gol67b], y se generan únicamente para determinadas combinaciones de los coeficientes de realimentación (véase el apéndice A). Nótese que para una misma función de realimentación $h(x)$ cambiar el valor inicial de las celdas de memoria sólo supone un desplazamiento o cambio de fase de la secuencia-m generada.

Las secuencias-m poseen muy buenas características pseudo-aleatorias, como demuestra que cumplan las tres condiciones mencionadas al inicio de este apartado. Su ACF periódica viene dada por (2.15), y no existe limite en cuanto a su longitud, que puede tomar cualquier valor $L = 2^N - 1$.

$$R_{m_i,m_i}[\tau] = \begin{cases} L, & \tau = 0 \\ -1, & \tau \neq 0 \end{cases} \tag{2.15}$$

Ahora bien, la CCF entre dos secuencias-m generadas a partir de distintos LFSRs proporciona, en general, resultados muy pobres. No obstante, para determinadas longitudes existen parejas de secuencias con buenas propiedades de correlación cruzada (véase la figura 2.5) que se conocen como *parejas preferidas*. Dos secuencias-m m_1 y m_2 constituyen una pareja preferida si:

- N no es divisible por 4.
- La secuencia m_2 puede obtenerse diezmando m_1 por un factor q ($m_2[n] = m_1[qn]$) de valor $q = 2^k + 1$ ó $q = 2^{2k} - 2^k + 1$.
- El máximo común divisor de N y k es $mcd(N, k) = \begin{cases} 1, & \text{si } N \text{ es impar} \\ 2, & \text{si } N \text{ es par} \end{cases}$
- La CCF periódica tiene tres valores distintos:

$$R_{m_1,m_2}(\tau) = \{-1, \ -t(N), \ t(N) - 2\}$$

$$t(N) = \begin{cases} 1 + 2^{\frac{N+1}{2}}, & \text{si } N \text{ es impar} \\ 1 + 2^{\frac{N+2}{2}}, & \text{si } N \text{ es par} \end{cases} \tag{2.16}$$

Es posible formar *conjuntos preferidos* de secuencias-m donde todas los parejas que se puedan establecer entre dichas secuencias sean preferidas. Sin embargo, el número M de secuencias-m que componen el conjunto es muy reducido, por lo que no se aconsejan para entornos multi-usuario. La tabla 2.2 recoge el número máximo de secuencias que componen

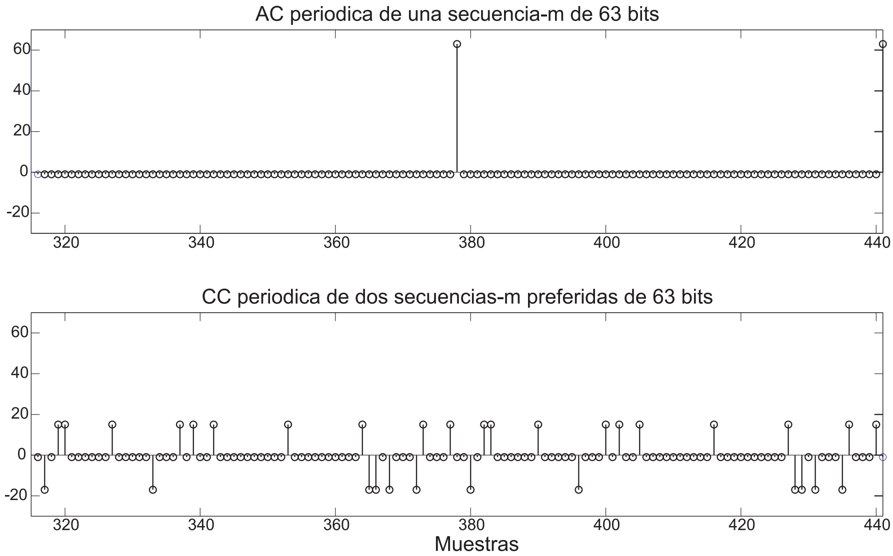


Figura 2.5: ACF y CCF periódicas de secuencias-m preferidas de 63 bits.

un conjunto preferido, así como sus cotas de correlación cruzada (θ_{CC}) periódica. Con objeto de poder establecer una comparativa, se incluye también el número de secuencias-m disponibles de una misma longitud (generadas a partir de distintos LFSRs), y su cota de correlación cruzada periódica[3] asociada. Un valor $M = 0$ indica que no existen secuencias-m preferidas de esa longitud.

N	L	Total de secuencias-m		Conjunto preferido	
		M	θ_{CC}	M	θ_{CC}
3	7	2	0.71	2	0.71
4	15	2	0.6	2	0.6
5	31	6	0.73	3	0.29
6	63	6	0.36	2	0.27
7	127	18	0.32	6	0.13
8	255	16	0.37	0	0.13
9	511	48	0.22	2	0.06
10	1023	60	0.37	3	0.06
11	2047	176	0.14	4	0.03
12	4095	144	0.34	0	0.03

Tabla 2.2: Número M de secuencias y cota de correlación cruzada (θ_{CC}), para todas las secuencias-m posibles de una determinada longitud L y para conjuntos preferidos.

[3]Los valores de correlación especificados en las ecuaciones (2.15) y (2.16), son válidos únicamente para el caso de emisiones periódicas, no así para aperiódicas.

Secuencias Gold

Las secuencias Gold [Gol67a] proporcionan un mayor número de secuencias con correlaciones cruzadas bajas que las disponibles usando parejas preferidas de secuencias-m, aunque sus valores de ACF empeoran.

El conjunto de códigos Gold asociado a una pareja preferida m_1 y m_2 de longitud $L = 2^N - 1$ está compuesto por dicha pareja preferida más la suma módulo-2 de la primera secuencia con cualquier versión desfasada de la segunda:

$$Gold = \{m_1, m_2, m_1 \oplus m_2, m_1 \oplus Dm_2, m_1 \oplus D^2 m_2, \cdots, m_1 \oplus D^{L-1} m_2\} \tag{2.17}$$

Donde $\oplus$ representa la operación suma en módulo-2 (XOR) y $D^k m_2$ es la secuencia que resulta de desplazar k posiciones cíclicamente la secuencia m_2. Así pues, cada conjunto de secuencias Gold está formado por un total de $M = 2^N + 1$ secuencias de período $L = 2^N - 1$. Una posible implementación de un generador de secuencias Gold se representa en la figura 2.6.

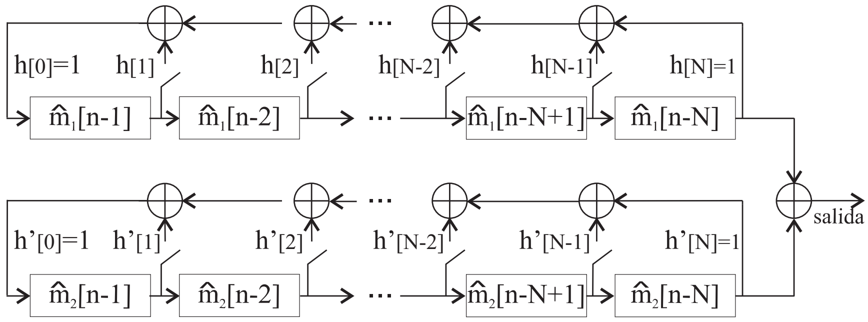


Figura 2.6: Generador de secuencias Gold.

A excepción de las secuencias m_1 y m_2 el resto no son secuencias de longitud máxima (su período es de $2^N - 1$ en lugar de $2^{2N} - 1$ como correspondería al usar dos LFSRs), por lo que su ACF empeora respecto a la conseguida con secuencias-m:

$$R_{gold_i, gold_i}[\tau] = \begin{cases} L, & \tau = 0 \\ \{-1, \ -t(N), \ t(N) - 2\}, & \tau \neq 0 \end{cases} \tag{2.18}$$

Los valores de CCF máximos obtenidos con las secuencias Gold son similares a los conseguidos con las secuencias-m, esto es, $\{-1, \ -t(N), \ t(N) - 2\}$. Un ejemplo de resultados de correlación obtenidos con secuencias Gold de 63 bits se muestra en la figura 2.7.

Resumiendo, la principal ventaja de las secuencias Gold frente a las secuencias-m es que disponen de un conjunto $M = 2^N + 1$ secuencias con CCF del mismo orden que las

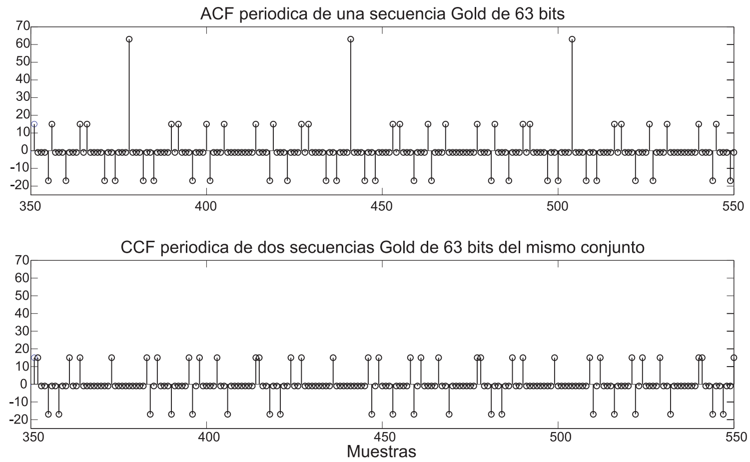


Figura 2.7: ACF y CCF periódicas de secuencias Gold de 63 bits.

obtenidas con las parejas preferidas. Sin embargo, son peores sus valores de ACF y no todas las secuencias Gold son balanceadas (la diferencia entre el número de $+1$ y -1 es mayor a la unidad; en concreto, es igual a $2^{\lfloor \frac{N+2}{2} \rfloor}+1$). Además, los valores de CCF periódica entre dos secuencias, aunque bajos, no son cero.

En [SP80] se muestra un conjunto de secuencias denominadas *Gold-like* por su parecido con las secuencias Gold, aunque se generan a partir de una secuencia diezmada. Si u es una secuencia-m de longitud $L = 2^N - 1$, siendo N par, y q es un entero tal que $mcd(q, L) = 3$, se forman nuevas secuencias v_k diezmando la secuencia $D^k u$, con $k = 0,\ 1\ y\ 2$, por el factor q. El conjunto de secuencias Gold-like está formado por u y la suma módulo dos de u con cualquier versión desfasada de v_k. En total se dispone de $M = 2^N$ secuencias, con cota de correlación igual a la de las secuencias Gold, esto es $\theta = t(N)/L$. Ahora bien, en lugar de tres se pueden obtener hasta cinco valores distintos en la ACF (para $\tau \neq 0$) y en la CCF: $\{-1,\ -t(N),\ t(N)-2,\ -s(N),\ s(N)+2\}$, siendo $s(N) = 1 + 2^{\frac{N}{2}}$ y N par.

Añadiendo un cero al final de las secuencias $\hat{m}_1$ y $\hat{m}_2$ se consigue un conjunto de secuencias Gold ortogonales [Tac92, DO99]. Teniendo en cuenta que los ceros se mapean como unos en la representación bipolar binaria, el nuevo conjunto quedaría como se indica en (2.19), donde la nueva longitud de las secuencias es $L' = 2^N$ y su CCF periódica en $\tau = 0$ es cero. Cuando el desfase entre las secuencias es mayor que el correspondiente a un bit, las funciones de correlación dejan de ser cero, y estas secuencias pierden su utilidad. Algo más tarde Saito et al. [SYKA98] realizan una búsqueda de aquellas secuencias dentro de un conjunto de códigos Gold ortogonales que presentan valores de correlación cero alrededor del origen. Esto es, buscan sub-conjuntos de secuencias Gold ortogonales generalizadas dentro de un conjunto Gold ortogonal. La tabla 2.3 muestra el número de secuencias encontradas para distintas longitudes de códigos Gold ortogonales, donde se puede apreciar la necesidad de recurrir a secuencias largas si se desea incrementar el tamaño de la IFW.

$$\begin{aligned} &Gold - O = \{m_1', m_2', m_1' \oplus m_2', m_1' \oplus Dm_2', m_1' \oplus D^2 m_2', \cdots, m_1' \oplus D^{L-1} m_2'\} \\ &m_1' = (m_1[0], m_1[1], \cdots, m_1[L-1], 1) \\ &m_2' = (m_2[0], m_2[1], \cdots, m_2[L-1], 1) \end{aligned} \tag{2.19}$$

Longitud	W=1	W=3	W=5	W=7	W=9	W=11
32	32	8	2	2	-	-
64	64	16	4	3	2	2
128	128	32	8	2	2	2

Tabla 2.3: Número de códigos Gold GO en función de su longitud (L') y el tamaño de la IFW ($2W + 1$) [SYKA98].

Secuencias Kasami

En función del número de secuencias que componen un conjunto Kasami se habla de conjunto pequeño o conjunto grande de secuencias Kasami, siendo distintas sus propiedades de correlación.

El conjunto pequeño de secuencias Kasami [Kas68] mejora la CCF de las secuencias Gold, aunque presenta un menor número de secuencias disponibles en un conjunto. Su mecanismo de generación es muy sencillo. Se parte de una secuencia-m m_1 de longitud $L = 2^N - 1$ obtenida a partir de un LFSR con un número par de celdas de memoria, esto es, $\mathrm{mod}(N, 2) = 0$. Una segunda secuencia de período $2^{\frac{N}{2}} - 1$ es generada a partir de diezmar la secuencia m_1 por un factor $q = 2^{\frac{N}{2}} + 1$. Concatenando q veces la secuencia diezmada se obtiene una nueva secuencia m_2 de longitud $L = 2^N - 1$. El conjunto pequeño de secuencias Kasami está formado por m_1 y la suma módulo-2 de m_1 con cualquier versión desfasada de m_2.

$$Kasami = \{m_1, m_1 \oplus m_2, m_1 \oplus Dm_2, m_1 \oplus D^2 m_2, \cdots, m_1 \oplus D^{L'-1} m_2\} \tag{2.20}$$

Donde $L' = 2^{\frac{N}{2}} - 1$ representa la longitud de la secuencia m_1 diezmada por el factor q. El conjunto pequeño de secuencias Kasami contiene por tanto $M = 2^{\frac{N}{2}}$ secuencias frente a las $2^N + 1$ que forman un conjunto Gold.

Las ACF y CCF periódicas de estas secuencias toman valores en el rango:

$$R_{k_1,k_2}[\tau] = \begin{cases} L, & \tau = 0, \quad k_1 = k_2 \\ \{-1, \ -s(N), s(N) - 2\}, & \tau \neq 0, \quad k_1 = k_2 \\ \{-1, \ -s(N), s(N) - 2\}, & \forall\tau, \quad\ \ k_1 \neq k_2 \end{cases} \tag{2.21}$$

En donde $s(N) = 1 + 2^{\frac{N}{2}}$. Según se observa en la expresión (2.21) los valores de correlación cruzada del conjunto pequeño de secuencias Kasami son del orden de la mitad de los obtenidos con las secuencias Gold y presentan además un mejor comportamiento en entornos asíncronos. Por otro lado, al igual que las Gold, no son balanceadas, siendo la diferencia máxima entre unos y menos unos igual a $2^{\frac{N}{2}}$.

La figura 2.8 muestra un ejemplo de ACF y CCF periódicas de dos secuencias de 63 bits pertenecientes al conjunto pequeño Kasami. Al igual que ocurre con las otras secuencias pseudoaleatorias detalladas anteriormente, cuando la emisión se realiza de modo aperiódico las propiedades de correlación se degradan. Un estudio de cómo afecta la aperiodicidad al conjunto pequeño de secuencias Kasami en términos de cotas de correlación se presenta en [Lah95].

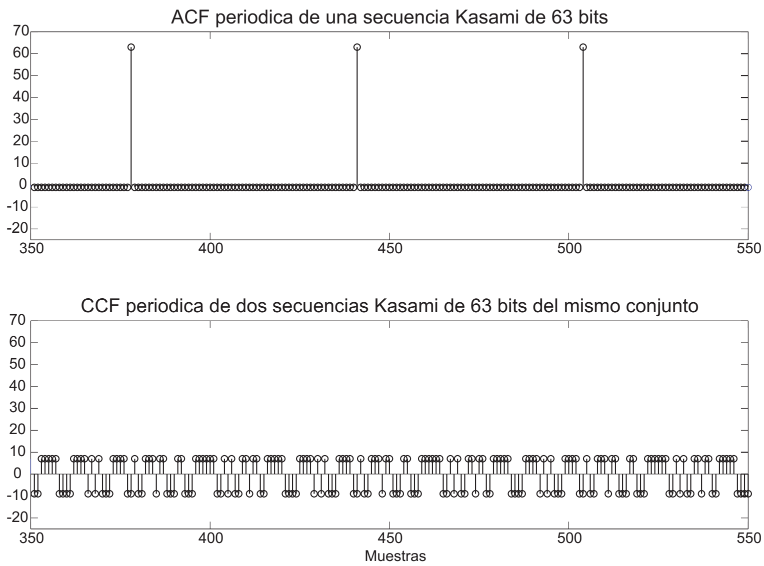


Figura 2.8: ACF y CCF periódicas de secuencias de 63 bits del conjunto pequeño Kasami.

El conjunto grande de secuencias Kasami es una combinación del conjunto pequeño y las secuencias Gold (o Gold-like) y, como su nombre indica, presenta un mayor número de secuencias que el anterior: $M = 2^{\frac{N}{2}}(2^N + 1)$ si se usan Gold o $M - 1$ en caso de usar Gold-like. Como contrapartida, la cota de correlación es la correspondiente a estas últimas secuencias $\theta = \frac{t(N)}{2}$, y la correlación puede tener cinco valores distintos $\{-1,\ -t(N),\ t(N)-2,\ -s(N),\ s(N)+2\}$. En definitiva la única ventaja de este conjunto es su elevado número de secuencias, pero no mejora los valores de CCF respecto a las secuencias-m y Gold, por lo que no se va a incidir más en su estudio. Para abreviar, de aquí en adelante se utilizará de modo genérico el nombre de secuencias Kasami haciendo referencia al conjunto pequeño de dichas secuencias.

2.2.4. Códigos Golay

Para evitar las interferencias ISI y MAI es necesario que los lóbulos laterales de la ACF y los valores de la CCF sean nulos. Usando códigos pseudo-aleatorios es posible tener bajas correlaciones cruzadas, pero con los conjuntos de M secuencias complementarias (M-CSS, *Complementary Set of M Sequences*) se consiguen correlaciones nulas sumando las funciones de correlación de las secuencias que componen dicho conjunto. Cuando el número de secuencias complementarias es $M = 2$ se obtienen parejas Golay de longitud $L = 2^a$, con $a \in \mathbb{N} - \{0\}$.

Estos códigos fueron descritos por primera vez por Marcel Golay [Gol49, Gol61] y son numerosas las aplicaciones ultrasónicas para la medida de TDV que hacen uso de ellas [HG88, DUG+99, DUM+99]. Se componen de un par de secuencias $a_0[n]$ y $a_1[n]$ de igual longitud L y valores $\{-1, 1\}$, siendo la suma de las ACF aperiódicas independientes de cada una de ellas una delta de Kronecker de peso $2L$:

$$C_{a_0,a_0}[\tau] + C_{a_1,a_1}[\tau] = 2L \cdot \delta[\tau] \tag{2.22}$$

La figura 2.9 muestra las propiedades de auto-correlación de un par Golay de 32 bits.

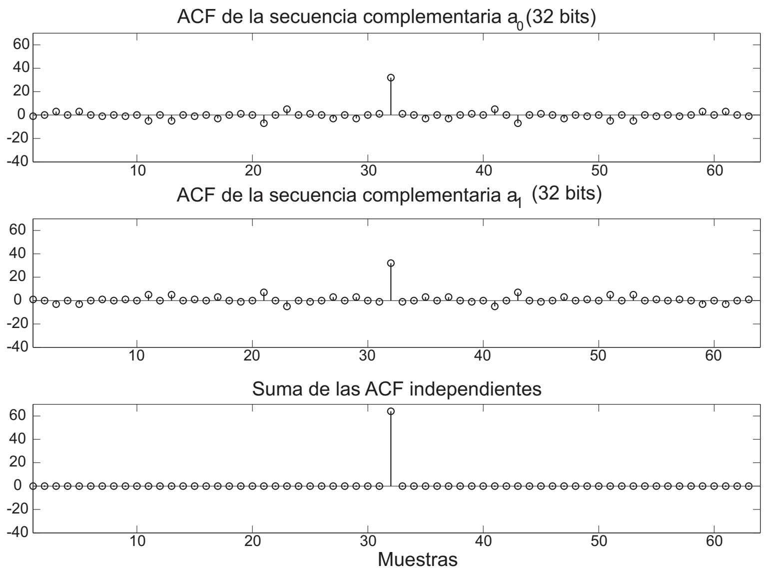


Figura 2.9: Ejemplo de ACF aperiódica de un par de secuencias Golay de 32 bits.

Existen parejas Golay de cualquier longitud $L = 2^a \cdot 10^b \cdot 26^c$, siendo a, b y c número enteros positivos. Budisin [Bud90a] propone un algoritmo recursivo que permite generar parejas Golay de cualquier longitud $L = 2^a$ partiendo de secuencias elementales delta. Este algoritmo constituye la base de un correlador eficiente [Bud91, Pop99] que reduce en gran medida el número de operaciones a realizar, y que puede ser fácilmente implementado en hardware configurable [HUH+03]. En consecuencia, las parejas Golay de longitud $L = 2^a$ son las más utilizadas.

Es posible además encontrar dos parejas Golay completamente incorreladas entre sí, es decir, la suma de las correlaciones entre las correspondientes secuencias de cada par es cero para cualquier desplazamiento, lo que en teoría hace posible la emisión simultánea de dos señales sin que haya interferencias entre ellas. Si $A = (a_0, a_1)$ y $B = (b_0, b_1)$ son dos parejas Golay cualesquiera con esta propiedad entonces:

$$C_{A,B}[\tau] = C_{a_0,b_0}[\tau] + C_{a_1,b_1}[\tau] = 0, \quad \forall\tau \tag{2.23}$$

Como curiosidad, en 1960 George R. Welti utilizaba parejas binarias de longitud par como base para la generación de un nuevo código cuaternario [Wel60]. Varios años más tarde Wilson demuestra que estos pares coinciden con los propuestos por Golay [WR79]. Entrelazando los bits de dichos pares se forma el código cuaternario propuesto por Welti. En el trabajo de Welti se describe además la estructura de un posible generador del código cuaternario y de su filtro acoplado asociado, requiriendo ambos un número de operaciones igual a $log_2(L)$. Estos códigos, sin embargo, no han alcanzado la difusión e interés que suscitan los pares Golay.

2.2.5. Conjuntos de secuencias complementarias

Los conjuntos de M secuencias complementarias (M-CSS) constituyen una generalización de los pares Golay, en donde el número de secuencias M que componen el conjunto puede ser mayor que dos [TL72]. Como ventaja frente a las Golay, los CSS disponen de hasta M conjuntos mutuamente incorrelados entre sí.

Un conjunto $S_i^{(M,L}$ con M secuencias binarias $s_{i,j}$ de longitud L, $\{S_i^{(M,L} = s_{i,j}^{(M,L}[l] \in \{-1, 1\}; 0 \leq i, l \leq L-1; 0 \leq j \leq M-1\}$, es complementario si la suma de sus funciones de auto-correlación (SACF) es nula para todos los desplazamientos distintos de cero, y tiene un valor máximo de $M \cdot L$ para el caso de un desplazamiento nulo.

$$SACF = \sum_{j=0}^{M-1} C_{s_{i,j},s_{i,j}}[\tau] = M \cdot L \cdot \delta[\tau] = M^{N+1} \cdot \delta[\tau] \tag{2.24}$$

En donde $\delta[\tau]$ es la función delta de Kronecker, y $L = M^N$ la longitud de las $M = 2^m$ secuencias que componen el conjunto, pudiendo ser m y N cualquier valor natural a excepción del cero[4]. Para simplificar se han omitido en la expresión anterior los superíndices

[4]La longitud L de las secuencias y el número de conjuntos S_i posibles dependen del mecanismo de generación utilizado. En esta tesis se ha hecho uso de los algoritmos recursivos propuestos en [DMUH+07b], lo que supone que $L = M^N$ y $S_i; 0 \leq i \leq L-1$. Otros posibles algoritmos, que derivan en CSS de distinta longitud y número de secuencias, son [SM88, CCY+05].

(M, L); de ahora en adelante sólo se usarán cuando sea necesario especificar un número M concreto de secuencias de una longitud L específica.

Dados dos CSS, S_i y $S_{i'}$, se dice que uno es compañero del otro si se cumplen las siguientes propiedades:

- Ambos conjuntos tienen el mismo número de secuencias, M en este caso, y todas tienen la misma longitud L.
- La suma de las funciones de correlación cruzada (SCCF) de las secuencias de ambos conjuntos es nula para cualquier desplazamiento:

$$SCCF = \sum_{j=0}^{M-1} C_{s_{i,j}, s_{i',j}}[\tau] = 0, \qquad \forall \tau \tag{2.25}$$

Varios CSS son mutuamente incorrelados si todos los pares de conjuntos que se puedan establecer entre ellos son compañeros el uno del otro. Está demostrado que, para cualquier CSS, el número de conjuntos mutuamente incorrelados (UCSS, *Uncorrelated Complementary Sets of Sequences*) no puede exceder el número de secuencias M del conjunto. En concreto, con el mecanismo de generación utilizado, se obtiene una familia de M-UCSS a partir de un conjunto S_i como: $\{S_{(i+i'\frac{L}{M}) \bmod (L)}; 0 \leq i' \leq M-1\}$ [DMUH+07b]. En el apéndice B se describe en detalle el mecanismo de generación de los CSS utilizado.

Por otro lado, trabajando en un entorno síncrono basta con que la correlación cruzada sea cero para $\tau = 0$, en cuyo caso, se utilizarán CSS mutuamente ortogonales (OCSS, *Orthogonal Complementary Sets of Sequences*). El número P de estos conjuntos OCSS es mucho mayor que el de UCSS, cumpliéndose: $P \leq M \cdot L$.

Además de secuencias complementarias binarias, existen secuencias complementarias con valores enteros [DK88], reales [Bud90b] e incluso complejos [Siv78]. Sin embargo, todas ellas están fuera de los objetivos de esta tesis.

2.2.6. Códigos LAS

En marzo de 2003 Daoben Li presenta los códigos LAS (*Large Area Synchronous*) como solución a los problemas derivados de usar códigos ortogonales en comunicaciones móviles cuasi-síncronas que utilizan técnicas de acceso al medio por división en el código (CDMA)[Li03]. En este mismo trabajo Li propone los códigos LAS como base para la telefonía móvil de cuarta generación.

Los códigos LAS son ortogonales generalizados, esto es, sus funciones de correlación tienen una ventana libre de interferencias (IFW) alrededor del origen. Así, es posible eliminar las ISI y MAI si la diferencia entre los instantes de llegada de las distintas recepciones, expresados en número de muestras, es menor que W, siendo W el tamaño de la semi-ventana libre de interferencias $IFW = 2W + 1$ (véase la figura 2.10).

Se basan en la combinación de los códigos LA (*Large Area*) y LS (*Loosely Synchronous*). El estándar LAS-2000 [LCLH04] para comunicaciones inalámbricas (compatible en la capa física con CDMA-2000) utiliza los códigos LS para realizar la separación entre usuarios de un mismo sector. Estos códigos son modulados por posición de pulso (PPM, *Pulse Position Modulation*) por el código LA, cuya utilidad reside en codificar los distintos sectores que componen el área a cubrir. El código LS se inserta en aquellas posiciones del código LA donde hay un bit o pulso positivo, y su versión negada se copia en donde se encuentran bits negativos, según se muestra en la figura 2.11.

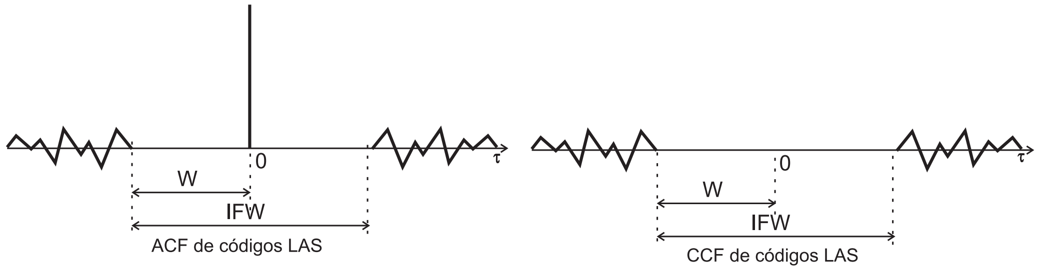


Figura 2.10: ACF y CCF de códigos LAS.

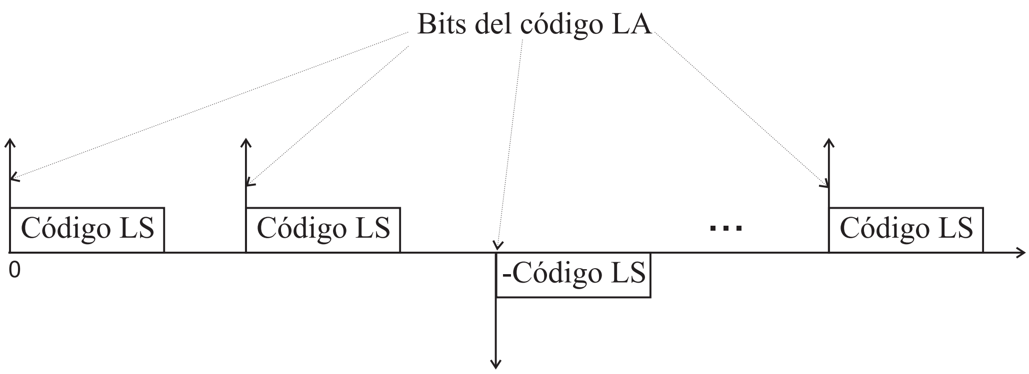


Figura 2.11: Construcción de códigos LAS a partir de códigos LA y LS.

Códigos LA

Los códigos LA están compuestos por elementos $\{-1, 0, 1\}$ y se generan insertando cadenas de ceros de distinta longitud entre los bits de cualquier conjunto de códigos ortogonales. No existe un único método para insertar las cadenas de ceros, aunque en todos los casos debe cumplirse [Li99]:

- Las cadenas de ceros se ordenan del mismo modo en las M secuencias del conjunto ortogonal.
- Todas las cadenas de ceros menos una deben tener longitud par.
- No puede haber dos cadenas de ceros de la misma longitud.
- Ninguna longitud o suma de las longitudes de las cadenas de ceros puede ser resultado de la suma de la longitud de las otras cadenas.
- La longitud L final de los códigos LA debe ser impar. Es por esto que si la longitud L_0 de los códigos ortogonales es par, una de las cadenas de ceros debe ser impar. Normalmente esta cadena impar tiene sólo un cero más que la cadena de menor longitud.

Los códigos LA resultantes tienen las siguientes propiedades de correlación:

- El valor del pico principal de la ACF es L_0, siendo L_0 la longitud de los códigos ortogonales (normalmente $L_0 = M$).
- Los lóbulos laterales de la ACF y los picos de CCF pueden tener sólo tres valores: -1, 0 y 1.
- Existe una zona sin interferencias de tamaño W alrededor del origen.

Estos códigos se suelen definir con tres parámetros $LA(L_0, NZ, L)$, en donde NZ es el tamaño de la cadena con menor número de ceros y L es la longitud de la secuencia LA. El número de códigos LA es igual al de códigos ortogonales $M = L_0$.

Como ejemplo, en la tabla 2.4 se muestra el número de ceros a insertar entre códigos ortogonales de 16 bits para obtener códigos LA de distinta longitud. Además en la figura 2.12 puede observase la ACF y CCF de códigos LA(4,2,21) generados a partir de códigos Walsh-Hadamard de orden 4.

Resulta evidente que el principal problema de estos códigos es su pobre relación $\frac{L_0}{L}$, esto es, el valor del pico principal de correlación es muy pequeño en relación a la longitud de la secuencia. Por ejemplo, utilizando códigos LA(16,50,1065) se obtiene un ratio $\frac{16}{1065} = 0.0150$, mientras que con las secuencias pseudo-aleatorias o CSS el ratio es siempre 1. Son, por tanto, códigos débiles ante interferencias, pudiendo quedar enmascarado fácilmente el pico principal de ACF en entornos ruidosos. Los códigos LAS mitigan este problema al combinar los códigos LA con los LS.

Códigos LS

Estos códigos presentan también una IFW alrededor del origen, y la relación $\frac{pico\ ACF}{L}$ es considerablemente mayor que la obtenida con códigos LA. Es por esto que se han considerado

NZ	L	Longitud de las cadenas de ceros
38	847	38 40 42 44 46 48 50 52 54 56 60 62 68 72 76 39
40	897	40 42 44 46 50 52 54 56 58 60 62 68 70 76 78 41
42	905	42 44 46 48 50 52 54 56 58 62 64 68 70 72 76 43
44	923	44 46 48 50 52 54 56 58 60 62 64 66 70 72 76 45
46	959	46 48 50 52 54 56 58 60 62 64 66 70 72 74 80 47
		46 48 50 52 54 56 58 60 62 64 66 70 72 74 78 47
⋮	⋮	⋮
84	1563	84 86 88 90 92 94 96 98 100 102 104 108 110 112 114 85
90	1653	90 92 94 96 98 100 102 104 106 108 110 112 114 116 120 91
96	1749	96 98 100 102 104 106 108 110 112 114 116 118 120 122 126 97
98	1783	98 100 102 104 106 108 110 112 114 116 118 120 122 126 128 99

Tabla 2.4: Longitud de las cadenas de ceros a insertar entre códigos ortogonales de $L_0 = 16$ bits para formar códigos LA.

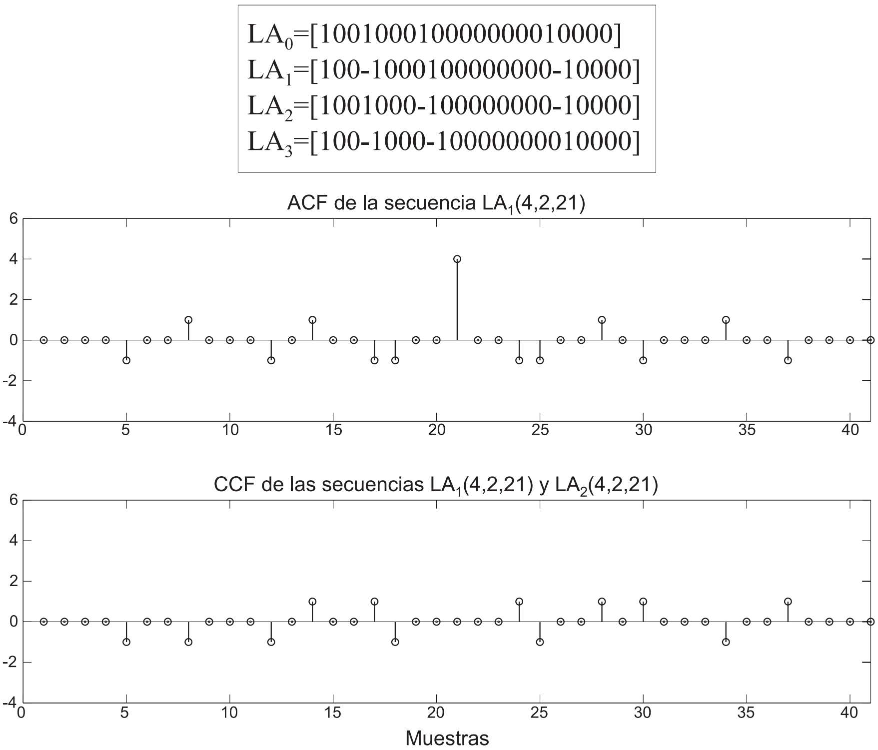


Figura 2.12: ACF y CCF de códigos LA(4,2,21).

únicamente los códigos LS, descartando los LA, en el estudio comparativo realizado en los siguientes capítulos.

Se generan concatenando secuencias de conjuntos complementarios. Específicamente, en [SBH01] se presenta un método para la generación recursiva de códigos LS a partir de dos parejas Golay incorreladas. Más tarde, Chao Zhang [ZYH05] propone un método más general a partir de conjuntos complementarios de cualquier número M de secuencias. Estos métodos son la base sobre la que se han construido los esquemas de correlación eficiente de códigos LS propuestos en esta tesis. Por ello, se ha considerado interesante incluir un breve resumen acerca de ambos esquemas de generación.

Un conjunto de $K = 2^n,\ n \in \mathbb{N}$, códigos LS de longitud L $\{G = g_k[l]; 0 \leq k \leq K-1; 0 \leq l \leq L-1\}$ compuesto por elementos $\{-1, 0, 1\}$ puede generarse a partir dos parejas Golay $S_0 = (s_{0,0}, s_{0,1})$ y $S_1 = (s_{1,0}, s_{1,1})$ de longitud L_0 e incorreladas entre sí, según la siguiente ecuación:

$$G_k[z] = \sum_{i=0}^{K/2-1} h_{k,i} \cdot z^{-i \cdot L_0} \cdot [S_{\pi_i,0}[z] + z^{-(\frac{K}{2}L_0+W)} S_{\pi_i,1}[z]] \tag{2.26}$$

$G_k[z]$ representa la transformada Z de $g_k[\tau]$; $h_{k,i} \in \{-1, 1\}$ son los elementos de una matriz de Hadamard de tamaño $(\frac{K}{2} \times \frac{K}{2})$; el vector $\Pi = [\pi_0, \pi_1, \cdots, \pi_{\frac{K}{2}-1}] \in \{0, 1\}$ denota la representación binaria de un número natural cualquiera $n,\ 0 \leq n < 2^{\frac{K}{2}}$, es decir, $n = \sum_i \pi_i 2^i$; y, finalmente, $W \leq L_0 - 1$ es el tamaño de la semi-ventana libre de interferencias (normalmente se usa $W = L_0 - 1$). Los primeros $K/2$ códigos LS se generan concatenando las parejas Golay con la polaridad indicada por los coeficientes $h_{k,i}$ de la matriz de Hadamard, y siguiendo el orden establecido por π_i. Para obtener los siguientes $K/2$ códigos, es necesario sustituir en la ecuación (2.26) el vector Π por su complementario $\Pi^* = [\pi_0^*, \pi_1^*, \cdots, \pi_{\frac{K}{2}-1}^*], \pi_k^* = \pi_k + 1(\mathrm{mod}\,2); 0 \leq k \leq \frac{K}{2} - 1$. Los códigos LS obtenidos tienen una longitud $L = KL_0 + W$; aunque, si las emisiones son periódicas, es necesario insertar un intervalo de guarda de, al menos, W ceros. En estos casos $L = KL_0 + 2W$, como queda ilustrado en la figura 2.13. El valor del pico máximo de auto-correlación es igual al número de elementos no nulos del código, esto es, $K \cdot L_0$.

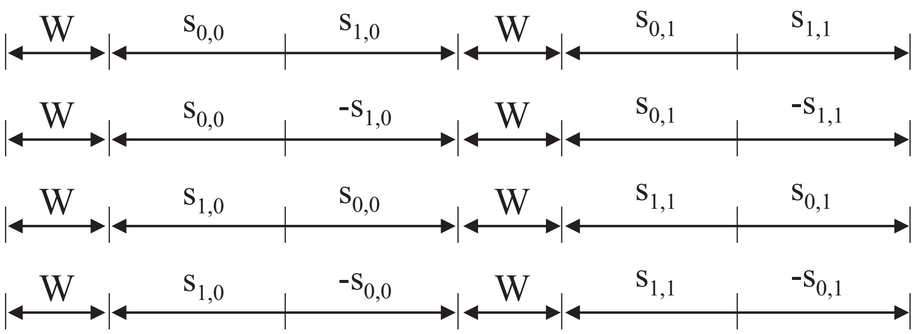


Figura 2.13: Estructura de $K = 4$ códigos LS, generados según [SBH01] a partir de un vector $\pi = [0\ 1]$ y una matriz de Hadamard $h_0 = [1\ 1],\ h_1 = [1\ -1]$.

Como se ha comentado anteriormente, es posible generar códigos LS a partir de M

conjuntos de secuencias complementarias de longitud L_0 incorrelados entre sí. En [ZYH05] se propone un algoritmo genérico que tras P iteraciones permite obtener $K = M^{(1+P)}$ códigos LS de longitud $L = M^{(1+P)} \cdot L_0 + (M-1)W$. El método anterior basado en pares Golay puede considerarse un caso especial de este algoritmo cuando $M = 2$. Ahora bien, incrementar el número P de iteraciones para incrementar el número de códigos o su longitud supone aumentar también la complejidad del esquema de generación, y de los esquemas de detección eficiente que puedan derivarse de estos códigos. Como solución, se ha establecido $P = 1$, incrementando la longitud y número de códigos a través del número M de secuencias del CSS. Sabiendo esto, si se forma una matriz de tamaño $M \times M \cdot L_0$ en donde cada fila representa un CSS incorrelado con los demás cuyas M secuencias complementarias se han colocado una detrás de otra como se indica en (2.27), es posible obtener códigos LS concatenando las secuencias que componen dicha matriz según (2.28).

$$\mathbf{S} = [s_{i,j}]_{M\times M\cdot L_0} = \begin{pmatrix} S_0 \\ S_1 \\ \vdots \\ S_{M-1} \end{pmatrix} = \begin{pmatrix} \underset{\substack{\uparrow \\ s_{0,0}[0]s_{0,0}[1]\cdots s_{0,0}[L_0-1]}}{\boxed{s_{0,0}}} & s_{0,1} & \cdots & s_{0,M-1} \\ s_{1,0} & s_{1,1} & \cdots & s_{1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{M-1,0} & s_{M-1,1} & \cdots & s_{M-1,M-1} \end{pmatrix} \tag{2.27}$$

$$G_{n,m}[z] = \sum_{i=0}^{M-1} h_{m,i} \cdot z^{-iL_0} \cdot \left[\sum_{j=0}^{M-1} z^{-j(ML_0+W)} S_{\pi_{n,i},j}[z] \right] \tag{2.28}$$

En este caso $G_{n,m}[z]$ representa la transformada Z de $\{g_{n,m}[\tau]; 0 \le n,m \le M-1\}$; $h_{m,i}$ son los elementos de una matriz de Hadamard de tamaño $M \times M$; $\pi_{n,i} = (n+i) \bmod M$; y $W \le L_0 - 1$. De nuevo, la polaridad de las secuencias complementarias depende del signo de los coeficientes $h_{m,i}$, y el vector $\{\pi_n = (n+i) \bmod M \in \{0,1,\cdots,M-1\}; 0 \le i,n \le M-1\}$ indica qué conjunto se considera en cada momento. Se obtienen así un total de $K = M^2$ códigos LS de longitud $L = M^2L_0 + (M-1)W$ y pico principal de auto-correlación M^2L_0. Cuando $M = 2$ y $K = 4$ el resultado de las ecuaciones (2.26) y (2.28) es el mismo.

De aquí en adelante cuando se haga referencia al método de generación a partir de CSS propuesto en [ZYH05] se asumirá $P = 1$ y la ecuación de generación (2.28). Por otro lado, se usará la nomenclatura $LS(M, L_0, K)$ para denotar un conjunto de K secuencias LS, generados a partir de M-CSS de tamaño L_0, asumiendo $W = L_0 - 1$. El método de generación, a partir de Golay o de CSS, se puede intuir fácilmente a través de los valores asignados a M y K.

En la figura 2.14 se ilustra la construcción de una secuencia LS a partir de cuatro 4-CSS incorrelados. Asimismo, la figura 2.15 muestra las ACF y CCF de los códigos LS(4,16,16) $g_{0,0}$ y $g_{0,1}$. En este caso, la relación $\frac{pico\ ACF}{L} = \frac{M^2 L_0}{L} = 0.8505$.

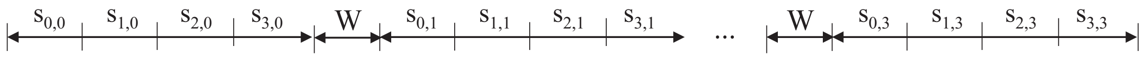


Figura 2.14: Estructura del código LS $g_{0,0}$ generado según [ZYH05] a partir de un vector $\pi_0 = [0\ 1\ 2\ 3]$ y la columna $h_0 = [1\ 1\ 1\ 1]$ de una matriz de Hadamard.

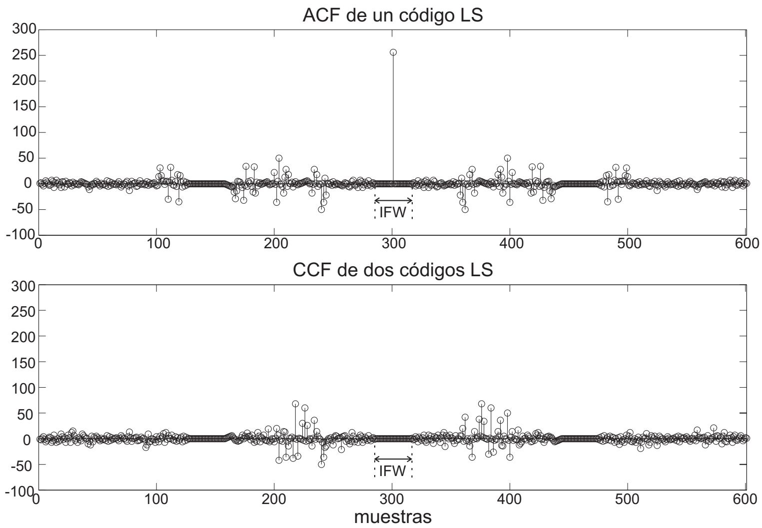


Figura 2.15: ACF y CCF de códigos LS de $L = 301$ bits, generados a partir de $M = 4$ CSS de longitud $L_0 = 16$.

Fuera de la IFW los valores de correlación que presentan los códigos LS son mayores que los obtenidos trabajando con secuencias pseudo-aleatorias o CSS. Es decir, en entornos completamente asíncronos las ISI y MAI de los códigos LS son más elevadas que las de códigos convencionales. De ahí la importancia de asegurar que la diferencia de tiempos máxima entre recepciones sea menor que W.

2.2.7. Códigos con tres zonas de correlación cero T-ZCZ

Los mismos autores que proponían la generación de códigos LS a partir de CSS, describen en el año 2004 un nuevo tipo de parejas denominadas T-ZCZ (*Three ZCZ codes*)[5] y cuya particularidad reside en que la suma de sus funciones de correlación aperiódica presenta tres zonas de correlación cero: alrededor del origen y en las dos terminaciones (véase la figura 2.16) [ZLH04, ZLH05].

[5] En las primeras referencias [ZLH04] los autores, con un poco de imaginación, encuentran cierta similitud entre las zonas de correlación cero laterales y las orejas de una persona, por lo que denominan a estos códigos *Ear Zero Correlation Zone codes* (E_{ZCZ}). Sin embargo, en trabajos posteriores [ZLH05] ya adoptan la terminología T-ZCZ (*Three Zero Correlation Zones*).

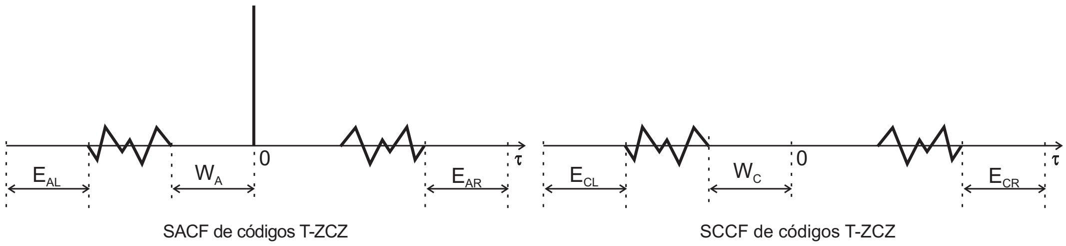


Figura 2.16: SACF y SCCF de códigos T-ZCZ.

Las parejas T-ZCZ presentan dos ventajas frente a los códigos LS. Por un lado no necesitan intervalos de guarda adicionales en caso de emisiones periódicas para mantener la IFW alrededor del origen. Por otro, presentan una mayor ganancia de proceso que las LS, siendo la relación $\frac{pico\ ACF}{L} = 2$. Este aumento de ganancia es, en parte obvio, puesto que al tratarse de pares de secuencias se transmite el doble de información que con las LS. Ahora bien, al no existir cadenas de ceros en el código, el pico principal de ACF de una única secuencia del par T-ZCZ es L, frente al pico de valor $K \cdot L_0 < L$ de las secuencias LS. Además, asumiendo una pequeña zona de interferencias en las funciones de correlación, los códigos T-ZCZ permiten realizar un mayor número de emisiones simultáneas que los pares Golay.

Un conjunto de M pares binarios $\{E_i = (e_{i,0}, e_{i,1}); 0 \leq i \leq M-1\}$ con longitud L y elementos $\{-1, 1\}$, es un conjunto T-ZCZ si la suma de sus auto-correlaciones (SACF) cumple (2.29) y la suma de sus correlaciones cruzadas (SCCF) cumple (2.30).

$$SACF[\tau] = C_{e_{i,0},e_{i,0}} + C_{e_{i,1},e_{i,1}} = \begin{cases} 2L & \tau = 0 \\ 0 & 1 \leq |\tau| \leq W_A \\ 0 & L - E_{AR} \leq \tau \leq L \\ 0 & -L \leq \tau \leq -(L - E_{AL}) \end{cases} \tag{2.29}$$

$$SCCF[\tau] = C_{e_{i,0},e_{i',0}} + C_{e_{i,1},e_{i',1}} = \begin{cases} 0 & 0 \leq |\tau| \leq W_C \\ 0 & L - E_{CR} \leq \tau \leq L \\ 0 & -L \leq \tau \leq -(L - E_{CL}) \end{cases} \tag{2.30}$$

Donde W_A es el tamaño de la semiventana libre de interferencias alrededor del origen de la SACF, y W_C el tamaño de la semi-ventana alrededor del origen de la SCCF; E_{AR} representa el tamaño de la zona libre de interferencias en el límite derecho de la SACF y E_{AL} el tamaño de dicha zona en el límite izquierdo; mientras que E_{CL} y E_{CR} representan la zona sin interferencias en los límites laterales izquierdo y derecho de la SCCF. Aunque las SACF y SCCF no tienen por qué ser simétricas, suele considerarse sólo el caso $W_A = W_C = E_{AL} = E_{C_L} = E_{AR} = E_{CR}$, hablando entonces de una zona libre de interferencias W genérica. Nótese que cuando $W = L/2$ los códigos T-ZCZ se reducen a pares Golay. La figura 2.17 muestra un ejemplo de las ACF de cada una de las secuencias que componen un par E_0 de longitud $L = 32$ bits, así como la suma de las mismas.

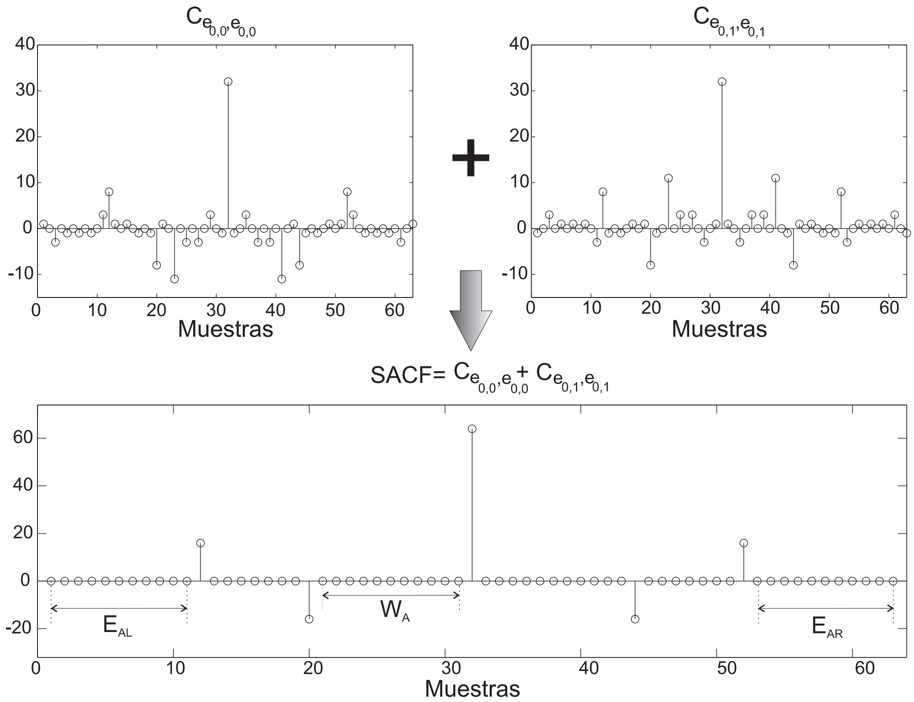


Figura 2.17: SACF de una pareja T-ZCZ de 32 bits.

Al igual que ocurría con los códigos LA y LS, los pares T-ZCZ pueden definirse mediante tres valores T-ZCZ(L,M,W). La longitud L de los pares, el número M de códigos que componen el conjunto y el tamaño W de las zonas libres de interferencias dependen del mecanismo de generación utilizado. En concreto hay tres mecanismos, que difieren en el modo en el que los bits de una matriz de generación Δ son considerados. Los detalles de dichos mecanismos de generación se describen en [ZLH04]. El algoritmo de generación de la matriz Δ es el mismo que el descrito por Tseng y Liu [TL72] para la generación de conjuntos complementarios incorrelados. Este algoritmo se explica en detalle en el apartado B.3 del apéndice B, no obstante, para facilitar la comprensión de los métodos de generación de los pares T-ZCZ se hace aquí un breve resumen.

Se parte de un par complementario $(s_{i,0}, s_{i,1})$ generado recursivamente según:

$$\begin{array}{rcl} s_{i,0}^{(0)} & = & s_{i,1}^{(0)} = 1 \\ s_{i,0}^{(m)} & = & s_{i,0}^{(m-1)} \mid \overline{s_{i,1}^{(m-1)}} \\ s_{i,1}^{(m)} & = & s_{i,1}^{(m-1)} \mid \overline{(-s_{i,0}^{(m-1)})} \end{array} \tag{2.31}$$

En donde $m \in \mathbb{N}$ es el número de iteración; $\overline{s_{i,0}}$ significa colocar en orden inverso los bits de $s_{i,0}$; y $s_{i,0} \mid s_{i,1}$ concatenar los bits de $s_{i,0}$ y $s_{i,1}$.

Se abre ahora un nuevo proceso de iteración para la construcción de una matriz Δ de generación con valor inicial:

$$\Delta^{(0)} = \begin{pmatrix} s_{i,0}^{(m)} & \overline{s_{i,1}^{(m)}} \\ s_{i,1}^{(m)} & -\overline{s_{i,0}^{(m)}} \end{pmatrix} \tag{2.32}$$

Después de n iteraciones el valor final de $\Delta^{(n)}$ es:

$$\Delta^{(n)} = \begin{pmatrix} \Delta^{(n-1)} \mid \Delta^{(n-1)} & -\Delta^{(n-1)} \mid \Delta^{(n-1)} \\ -\Delta^{(n-1)} \mid \Delta^{(n-1)} & \Delta^{(n-1)} \mid \Delta^{(n-1)} \end{pmatrix} \tag{2.33}$$

Haciendo uso de la matriz $\Delta^{(n)}$ se pueden conseguir pares T-ZCZ aplicando los métodos que se exponen a continuación.

Primer método. Se reescribe la matriz $\Delta^{(n)}$ de dimensiones $2^{n+1} \times 2^{2n+m+1}$ como:

$$\Delta^{(n)} = \begin{pmatrix} A_0^{(n)} \\ A_1^{(n)} \\ \vdots \\ A_{2^{n+1}-1}^{(n)} \end{pmatrix} \tag{2.34}$$

Donde $A_i^{(n)}$ es una fila de la matriz $\Delta^{(n)}$. Los pares $(A_0^{(n)}, A_1^{(n)})$, $(A_2^{(n)}, A_3^{(n)})$, $\cdots$, $(A_{2^{n+1}-2}^{(n)}, A_{2^{n+1}-1}^{(n)})$, y los pares que son compañeros a los anteriores $(\overline{A_1^{(n)}}, -\overline{A_0^{(n)}})$, $(\overline{A_3^{(n)}}, -\overline{A_2^{(n)}})$, $\cdots$, $(\overline{A_{2^{n+1}-1}^{(n)}}, -\overline{A_{2^{n+1}-2}^{(n)}})$ forman un conjunto de códigos T-ZCZ(L,M,W) con las siguientes características:

$$\begin{cases} \text{T-ZCZ}(2^{2n+1}, 2^{n+1}, 2^n) & m = 0 \\ \text{T-ZCZ}(2^{2n+m+1}, 2^{n+1}, 2^{n+m} + 2m - 1) & m > 0 \end{cases} \tag{2.35}$$

Segundo método. En este caso se divide la matriz $\Delta^{(n)}$ en dos sub-matrices del mismo tamaño $2^{n+1} \times 2^{2n+m}$. La sub-matriz $\Delta_L^{(n)}$ se corresponde con los valores de la mitad izquierda de la matriz $\Delta^{(n)}$, y la sub-matriz $\Delta_R^{(n)}$ con los del lado derecho, esto es, $\Delta^{(n)} = [\Delta_L^{(n)} \mid \Delta_R^{(n)}]$.

$$\Delta^{(n)} = \begin{pmatrix} A_{0L}^{(n)} & A_{0R}^{(n)} \\ A_{1L}^{(n)} & A_{1R}^{(n)} \\ \vdots & \vdots \\ A_{2^{n+1}-1L}^{(n)} & A_{2^{n+1}-1R}^{(n)} \end{pmatrix} \tag{2.36}$$

Los pares $(A_{0L}^{(n)}, A_{0R}^{(n)})$, $(A_{1L}^{(n)}, A_{1R}^{(n)})$, $\cdots$, $(A_{2^{n+1}-1L}^{(n)}, A_{2^{n+1}-1R}^{(n)})$ constituyen un conjunto T-ZCZ de valores:

$$\text{T-ZCZ}(2^{2n+m}, 2^{n+1}, 2^{n+m-1}) \tag{2.37}$$

Tercer método. La matriz $\Delta^{(n)}$ se divide en dos mitades derecha e izquierda ($\Delta_L^{(n)}$ y $\Delta_R^{(n)}$) según se comentó en el segundo método. Posteriormente se reescribe la matriz $\Delta_L^{(n)}$ como:

$$\Delta_L^{(n)} = \begin{pmatrix} A_{0,0}^{(n)} & A_{0,1}^{(n)} \\ A_{1,0}^{(n)} & A_{1,1}^{(n)} \\ \vdots & \vdots \\ A_{2^{2n+1}-1,0}^{(n)} & A_{2^{2n+1}-1,1}^{(n)} \end{pmatrix} \tag{2.38}$$

En donde cada una de los vectores $A_{i,0}^{(n)}$ y $A_{i,1}^{(n)}$ de tamaño $1 \times 2^{2n+m-1}$ son particiones del mismo tamaño de la matriz $\Delta_L^{(n)}$. Los pares $(A_{0,0}^{(n)}, A_{0,1}^{(n)})$, $(A_{1,0}^{(n)}, A_{1,1}^{(n)})$, $\cdots$, $(A_{2^{n+1},0}^{(n)}, A_{2^{n+1},1}^{(n)})$ son un conjunto T-ZCZ definido como:

$$\begin{cases} \text{T-ZCZ}(2^{m+1}, 4, 2^m - 1) & n = 1 \\ \text{T-ZCZ}(2^{2n+m-1}, 2^{n+1}, 2^{n+m-2}) & n > 1 \end{cases} \tag{2.39}$$

Si se utiliza Δ_R en lugar de Δ_L se obtienen los mismos resultados.

2.3. Revisión de trabajos sobre la aplicación de técnicas de codificación a la señal ultrasónica

2.3.1. Trabajos iniciales empleando sistemas analógicos

Robert Waag [WMRR72] es uno de los primeros autores en aplicar codificación a señales ultrasónicas, en este caso para la medida no invasiva del flujo sanguíneo en la cámara del corazón. En este trabajo se apuntan varias consideraciones interesantes. Por un lado recomienda la codificación de la señal emitida para facilitar su detección en el caso de trabajar en entornos ruidosos, y enfatiza la necesidad de caracterizar con precisión la señal emitida. Por otro lado, establece que dicha caracterización debe contemplar las cualidades de implementación del filtro acoplado a dicha secuencia. Los códigos elegidos son señales pseudo-aleatorias binarias que modulan en fase una portadora sinusoidal.

Otro de los primeros campos en emplear estas técnicas es el de la evaluación no destructiva de materiales. En un inicio, los trabajos de [FNBC75] y [BFN76] proponen la emisión de un ruido blanco gaussiano a través de un transductor ultrasónico. Los ecos reflejados en las fallas del material son capturados por el mismo transductor y correlados con una versión retardada de la señal emitida. El retardo se consigue propagando el ruido blanco emitido a través de un baño de agua con dos transductores, un emisor y un receptor. De este modo, en función de la separación entre los transductores se obtienen distintos retardos de la señal patrón. Cuando alguno de estos retardos coincide con el TDV de la señal emitida se produce un pico de correlación, indicando la profundidad a la que se encuentra un defecto en el material. Se demuestra en estos trabajos que un ensanchamiento del ancho de banda de la señal emitida viene acompañado de un incremento de la resolución espacial obtenida, y por tanto, un ruido blanco gaussiano es más eficiente que los impulsos de corta duración y gran energía que se venían usando hasta el momento. Ahora bien, el hecho de trabajar con secuencias completamente aleatorias implica depender de la línea de retardo para disponer de la secuencia original retardada como patrón en el proceso de correlación. Esta línea de retardo es voluminosa, no permite cambios suficientemente rápidos del retardo de la señal patrón y, lo que es peor, debido a la presencia de los dos transductores en el baño de agua, el ancho de banda y la resolución del sistema se reducen considerablemente.

Como solución, Charles M. Elias [EM78, Eli80] propone emitir un código pseudoaleatorio, en concreto una secuencia-m, cuyas características espectrales se aproximan a las del ruido blanco gaussiano y puede generarse de forma determinista, lo que permite prescindir de la línea de retardo del sistema anterior. Se utilizan en este caso dos registros de desplazamiento con la misma realimentación y valores iniciales para generar dos secuencias-m idénticas. La longitud de las secuencias ha sido elegida de modo que el pico de autocorrelación obtenido sea lo suficientemente elevado como para ser detectado en un entorno ruidoso. Ahora bien, al usar un único transductor para emisión y recepción se impone el uso de secuencias de menor longitud, puesto que durante la emisión el sistema de recepción está deshabilitado y defectos del material próximos al transductor no serían detectados. Por ello, las secuencias se transmiten en ráfagas de N bits, hasta completar la longitud total de las mismas. Las ráfagas correspondientes a la secuencia a emitir a través del medio se repiten con una separación fija, mientras que las correspondientes a la secuencia patrón lo hacen con un retardo que varía en pequeños incrementos. La primera ráfaga de la secuencia patrón se retarda de modo que es recibida antes que el eco correspondiente a la secuencia emitida, la siguiente se retarda el valor anterior más un incremento predeterminado, y así sucesivamente. La salida del correlador es resultado del producto analógico de la señal recibida con la secuencia patrón, después de un filtro paso bajo que actúa como integrador y una etapa de amplificación (véase la figura 2.18), obteniéndose un nuevo valor de correlación con cada nueva ráfaga de la secuencia-m. La división de la secuencia-m en sub-secuencias de menor tamaño implica una pérdida de las propiedades ideales de correlación de las secuencias-m, apareciendo lóbulos laterales. Para reducir el impacto de estos lóbulos, se propone dejar fijo el retardo de la señal patrón durante la emisión de un conjunto de ráfagas, obteniendo así una mejora en la relación pico principal de correlación a pico lateral.

Hasta el momento los trabajos mencionados usan un único emisor aislado. Aumentar el número de emisiones simultáneas desde distintos puntos del espacio de trabajo implica la obtención de una mayor información del entorno, lo que a su vez conlleva un incremento

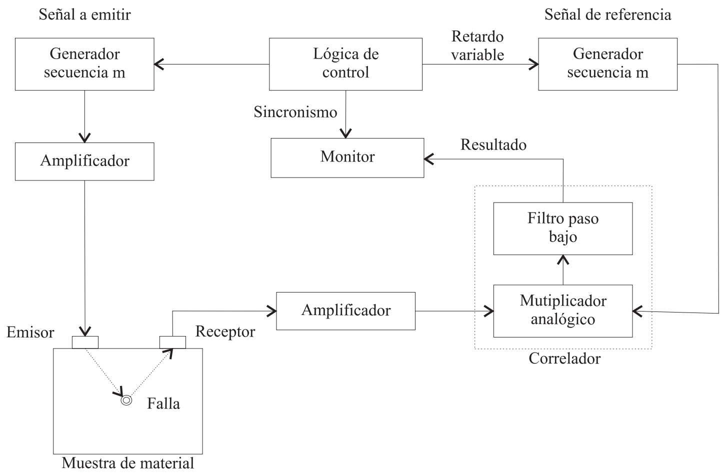


Figura 2.18: Sistema de evaluación no destructiva mediante emisión de secuencias-m propuesto por [EM78].

de la probabilidad de detección de posibles reflectores e incluso su clasificación. Aparece, no obstante, el problema de identificar correctamente el eco de cada emisor. Tournois [Tou80] plantea dos soluciones: emitir una misma señal a distintas frecuencias; o usar la misma frecuencia en todos los transductores, pero asignando a cada uno de ellos un código distinto e incorrelado con los demás. La primera solución se denomina FDMA (*Frequency Division Multiple Access*) y requiere un elevado ancho de banda, sólo disponible con transductores especiales y de gran coste. La segunda es conocida como CDMA (*Code Division Multiple Access*) y permite un uso más eficiente del espectro, un mayor número de emisores simultáneos y mejorar aspectos tales como la resolución espacial y relación señal-ruido del sistema. No se especifican en este trabajo las secuencias con las que codificar la señal ultrasónica. En el mismo simposio en el que se presentó este trabajo, se encuentra el realizado por Lee y Furgason [LF80] en donde se comparan las prestaciones de muestras de ruido, códigos Barker, secuencias-m y pares Golay en un sistema ultrasónico en el que tienen lugar dos emisiones simultáneas. Se descartan los códigos Barker debido a su escasa longitud (13 bits), y las muestras de ruido por ser las que mayores lóbulos laterales presentan en la ACF. Los menores lóbulos laterales se obtuvieron con los pares Golay; aunque se matiza que, en aquellas situaciones donde no es posible obtener la suma de las correlaciones de las dos secuencias que componen el par, los resultados son similares a los conseguidos con secuencias-m. Lo mismo ocurre para longitudes de código muy elevadas y transmisiones periódicas. En un trabajo posterior, los mismos autores [LF82] proponen la emisión simultánea de cuatro pares Golay y evalúan los lóbulos laterales obtenidos en función de la longitud de los códigos empleados.

2.3.2. Trabajos empleando sistemas digitales

Los trabajos revisados hasta ahora realizan la correlación mediante sistemas analógicos como el mostrado en la figura 2.18. En estos correladores es difícil controlar el inicio y final del intervalo de integración lo que conlleva la aparición de un ruido auto-inducido. Con la llegada de las tecnologías VLSI (*Very Large Scale Integration*) y VHSIC (*Very High Speed Integrated Circuit*) la implementación de correladores digitales comienza a ser viable, lo que deriva en resultados más precisos, proporcionados a mayor velocidad y más inmunes al ruido aleatorio. Además, se obtienen beneficios adicionales en términos de flexibilidad, diseño y tiempo de desarrollo, a costa de asumir un cierto error de cuantificación. Hayward y Gorfu [HG88] son de los primeros en utilizar correladores digitales para la evaluación no destructiva de materiales mediante señales ultrasónicas. Para la codificación de la señal ultrasónica usan un par Golay de 32 bits, que es emitido en banda base. La anchura de los pulsos se ha elegido de modo que las componentes frecuenciales de la señal están justo por encima de la frecuencia de operación de los transductores. En cuanto a la longitud de la señal, es suficiente como para tener una relación señal-ruido adecuada sin requerir un tiempo de proceso elevado.

En el ámbito del sónar aéreo, Peremans et al. [PAC93] proponen un sistema de clasificación de obstáculos en el que la señal ultrasónica emitida se codifica con un código Barker de 13 bits. El sistema está formado por un emisor/receptor y dos receptores laterales situados a 15 cm del anterior. Los TDV obtenidos simultáneamente en los tres transductores permiten discriminar entre reflectores planos y salientes, así como determinar su radio de curvatura. El código Barker es emitido empleando una modulación BPSK (*Binary Phase Shift Keying*) con un símbolo compuesto por cuatro ciclos de una portadora de 50 kHz. Estos códigos poseen lóbulos laterales cuyo valor es 1/13 el valor del lóbulo principal. Aparecen además interferencias consecuencia del proceso de demodulación. Esto significa que, aun cuando no haya fuentes de ruido, aparecerán en el resultado de correlación interferencias debido a este ruido auto-inducido. Para distinguir correctamente el pico de auto-correlación de las citadas interferencias, los autores añaden un detector de máximos locales. Cabe mencionar que, en este trabajo, el procesamiento de las señales recibidas es realizado *off-line*. Un esquema de codificación basado también en el uso de secuencias Barker es el empleado por Ureña [Ure98] en el desarrollo de un sistema sónar para la automatización de un vehículo industrial. En este caso la implementación del sistema de correlación se lleva a cabo en una FPGA (*Field Programmable Gate Array*) de bajo coste y se procesa el eco recibido en tiempo real.

En [JB98] se justifica la necesidad de multi-emisión en sistemas sónar sobre plataformas móviles, y se comentan los problemas que aparecen como consecuencia de la diafonía o *crosstalk*[6] entre emisores. Como solución a estos problemas, y para el caso de dos emisiones

[6]Se entiende como *diafonía o crosstalk* al fenómeno mediante el cual la señal emitida por un transductor es recibida en otro que está esperando su propio eco.

simultáneas, proponen transmitir dos secuencias pseudo-aleatorias de baja correlación cruzada cuya elección se ha realizado de forma experimental según se indica en [JB96]. Las secuencias se emiten en banda base, correlando la señal recibida con un eco real producido por un reflector situado a corta distancia. Los transductores hacen las veces de emisores y receptores, controlando ambos procesos mediante un DSP (*Digital Signal Processor*).

En [SB01] se propone sustituir los códigos pseudo-aleatorios del sistema anterior por conjuntos de pulsos, de modo que cada emisor tenga asociado un número de pulsos distinto a los demás. Cada receptor cuenta el número de pulsos recibidos, identificando así la fuente emisora. Después, mediante técnicas de triangulación se determina la posición del reflector. Aunque la implementación de este sistema es más simple y rápida que la del propuesto en [JB98], proporciona medidas erróneas cuando varios ecos se superponen. Esto conlleva que dos transductores adyacentes no puedan emitir simultáneamente, disminuyendo por tanto la velocidad de transmisión lo que puede resultar problemático en entornos muy dinámicos. Además, dos objetos próximos entre sí no serían detectados correctamente. Otro trabajo que no usa codificación para separar emisiones procedentes de fuentes independientes de ultrasonidos es el de [Kle99]. En este caso, los emisores transmiten dos pulsos de corta duración separados por un intervalo de tiempo que identifica cada emisor. Para validar un eco se comprueba el retardo relativo entre pulsos recibidos, considerando un posible desplazamiento Doppler en la separación de ambos pulsos como consecuencia del movimiento del obstáculo o del robot donde se encuentran instalados los transductores (tolera velocidades de hasta 1.3 m/s). Nuevamente, la velocidad de procesamiento es mayor que la requerida con técnicas de correlación. No obstante, el sistema sólo opera correctamente cuando la relación señal-ruido es elevada y si no se reciben pulsos superpuestos.

Los pares de secuencias complementarias Golay también se han usado en la codificación de la señal del sónar aéreo. Algunos trabajos a destacar han sido desarrollados en el Departamento de Electrónica de la Universidad de Alcalá. En [DUG+99] se propone emitir consecutivamente las dos secuencias del par, introduciendo un número de ceros determinado entre ellas, y modulándolas en BPSK. Un esquema parecido es planteado por los mismos autores en [DUM+99], con la salvedad de que los bits de ambas secuencias son entrelazados y emitidos secuencialmente. De este modo cambios del entorno afectan por igual a las dos secuencias del par. Otra propuesta de emisión aparece en [DUG+00, Her03], donde se utiliza una modulación QPSK (*Quadrature Phase Shift Keying*) digital para realizar la transmisión simultánea de las dos secuencias que forman el par Golay. Se consigue así reducir a la mitad el tiempo requerido por una modulación BPSK.

Otros trabajos en el mismo Departamento han utilizado la técnica de compresión de pulsos mediante secuencias Golay para otros fines. Como ejemplo, en [DUM+06] se detecta el paso de trenes en puntos específicos de la vía utilizando un array sensorial compuesto por pares de bobinas situadas a ambos lados del raíl, y excitadas mediante señales codificadas con secuencias Golay. Gracias a las propiedades de correlación de dichas secuencias el sistema puede trabajar en condiciones de señal-ruido adversas, e incluso cuando las señales se reciben muy atenuadas. En [GUH+04] se propone un sistema para la detección de objetos en vías ferroviarias, compuesto por una barrera con emisores de infrarrojos situada a un lado de la vía, y otra barrera con receptores enfrentada a la anterior en el otro lado de la vía. La presencia de obstáculos se detecta a partir de la interrupción de los enlaces entre emisores y receptores. Las emisiones se codifican con pares Golay distintos, transmitiendo simultáneamente las dos secuencias del par gracias a una modulación 4-PPM.

Volviendo a las aplicaciones en sónar aéreo, en [ÁUM+06] se propone un sistema sensorial para ambientes externos, capaz de operar en condiciones meteorológicas adversas. La particularidad de este sistema reside en el empleo de conjuntos de cuatro secuencias complementarias, lo que confiere al sistema la capacidad de aumentar a cuatro el número de emisiones simultáneas sin interferencia (frente a las dos conseguidas con los trabajos revisados hasta el momento). La figura 2.19 muestra el diagrama de bloques del sistema propuesto. A cada emisor se asocia un conjunto de secuencias distinto, cuyos bits se entrelazan y modulan en BPSK. Los emisores emiten simultáneamente sus códigos correspondientes y, en cada bloque de recepción, la señal recibida se demodula y correla con los cuatro conjuntos emitidos, obteniéndose picos en aquellas salidas asociadas a conjuntos que están presentes en la señal. Un esquema sensorial similar, pero utilizando conjuntos de ocho secuencias, se presenta en [DMUH+05].

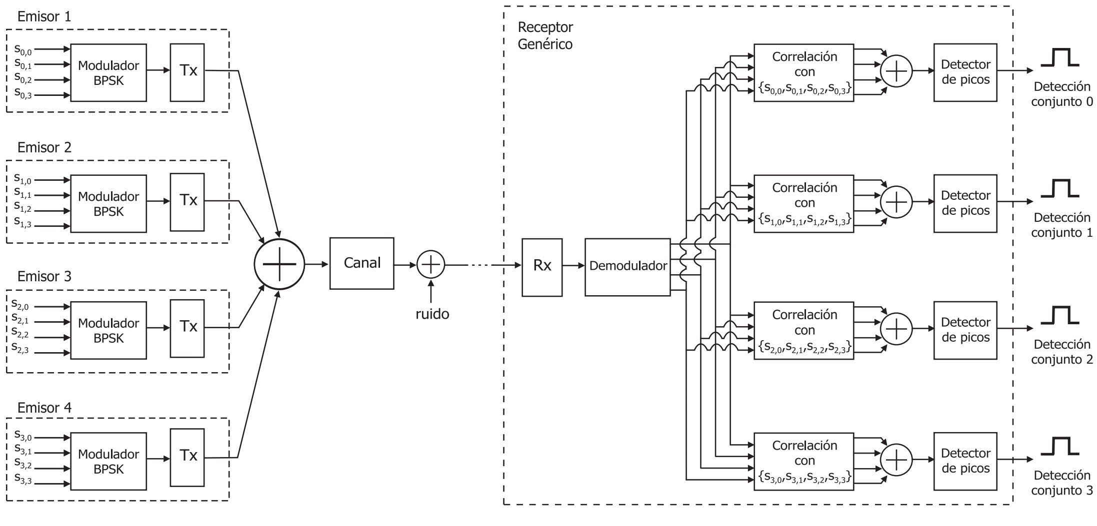


Figura 2.19: Esquema de codificación del sistema sensorial ultrasónico propuesto por [ÁUM+06].

2.3.3. Aplicación de técnicas de codificación a sistemas de posicionamiento basados en ultrasonidos

Los sistemas de posicionamiento local (LPS, *Local Positioning System*) pueden hacer uso de estas técnicas de compresión de pulsos consiguiendo así mejoras importantes. El objetivo final de estos sistemas es el cálculo de la posición absoluta de un objeto en un entorno cerrado y definido. Para ello, distintos autores proponen distintas soluciones, de entre las que no usan codificación destacan: el sistema *Bat* [HHWW99] desarrollado por la empresa AT&T y que se compone de una serie de usuarios móviles, cada uno portando un emisor de ultrasonidos, y un conjunto de balizas receptoras fijas en el techo. Un controlador central se encarga de enviar una señal de radio-frecuencia (RF) que dispara las emisiones, calculándose la posición de modo centralizado mediante un algoritmo de multilateración esférica[7]. Las precisiones obtenidas son del orden de 3 cm en el 95 % de los casos. Otro sistema a destacar es el *Cricket* [PMBT01]. En este trabajo son las balizas del techo las que actúan como emisoras de la señal ultrasónica, emitiendo asimismo un pulso de RF para la sincronización con el receptor móvil. En cada momento sólo una baliza puede emitir, evitando de este modo interferencias entre emisiones, lo que conlleva como contrapartida una baja frecuencia de operación. La posición se calcula en el móvil de forma local, asegurando la privacidad de la información. Los errores cometidos con este sistema oscilan entre los 5 y los 25 cm. Un trabajo similar es el presentado por Randell y Muller en [RM01], en donde la posición se obtiene con precisiones entre 10 y 25 cm.

Los sistemas descritos en el párrafo anterior utilizan transductores muy resonantes, con un ancho de banda de pocos kilohercios, lo que obliga a que todos los transductores emitan la misma señal ultrasónica, existiendo un problema de correspondencia de ecos. Para evitarlo se realiza una emisión secuencial, disminuyendo considerablemente la frecuencia de operación, lo que a su vez implica un mal comportamiento en medidas dinámicas. En [PMBT01] se trata de mejorar el comportamiento dinámico utilizando estimadores tipo Kalman para eliminar errores impulsivos. Además de estar limitado el acceso al medio, estos sistemas son muy sensibles al ruido ultrasónico y existe una falta de identificación de la señal emitida. La solución a estos problemas pasa por utilizar transductores de mayor ancho de banda y codificar las señales emitidas con códigos incorrelados.

En esta línea, las unidades *Dolphin* [HH06] desarrolladas en la Universidad de Cambridge utilizan transductores con un elevado ancho de banda, construidos a partir un material piezoeléctrico PVDF (Polifluoruro de Vinilideno), y codifican las emisiones con códigos pseudo-aleatorios Gold de 511 bits. Estos códigos son emitidos simultáneamente empleando una modulación BPSK con una portadora de 50 kHz. En la recepción se realiza la correlación con los códigos Gold enviados identificando así la fuente de emisión. A pesar de que la correlación cruzada entre los códigos Gold elegidos se encuentra dentro del margen tolerable por el sistema, aparecen otras interferencias propias de los sistemas CDMA, como son las causadas por el efecto cerca-lejos. Para mitigar estos efectos, se propone en este trabajo emplear métodos de cancelación por sustracción de interferencias (SIC) [Vit90]. Un receptor que emplea SIC resta las señales emitidas de la señal recibida a medida que

[7] Cuando existe sincronismo entre las balizas fijas y las unidades móviles, se calcula la posición mediante *multilateración esférica* midiendo distancias absolutas; mientras que si no existe sincronismo, se emplea *multilateración hiperbólica* a partir de diferencias de distancias.

se identifican, mejorando de este modo la detección del resto de emisores. No obstante, este método tiene asociado la dificultad de estimar correctamente la amplitud de la señal que debe ser restada. En cuanto a la disposición de emisores y receptores, se proponen dos configuraciones. En una de ellas, ocho receptores se fijan en el techo, cuatro emisores en puntos próximos a las paredes de la habitación y un quinto emisor tiene libertad de movimientos. La posición se calcula de forma centralizada en un ordenador personal, obteniendo medidas con una precisión de 2.3 cm con un nivel de confianza del 95 % [HW02]. La otra configuración se orienta a la privacidad, de modo que la unidad móvil actúa como receptora calculando su posición a partir de las señales que recibe de siete emisores emplazados en el techo. Si el receptor está sincronizado con las unidades emisoras se consiguen precisiones de 4.9 cm, y si no existe sincronismo la precisión disminuye hasta los 26.6 cm, en ambos casos con un nivel de confianza del 95 % [HW03].

Un trabajo similar, orientado a privacidad (el nodo móvil es el único que conoce su posición) y con recepción asíncrona, es el desarrollado entre la Universidad de Alcalá y el Instituto de Automática Industrial del CSIC [UHJ+07] (figura 2.20). En este caso las señales ultrasónicas se codifican mediante códigos Kasami de 255 bits, dichos códigos son modulados en BPSK con un símbolo formado por dos ciclos de una portadora de 50 kHz. El robot móvil detecta las señales de modo asíncrono, las demodula y correla con los códigos emitidos, calculando su posición mediante técnicas de trilateración hiperbólica a partir de las diferencias de tiempo de vuelo de los códigos recibidos. El procesamiento asociado a cada baliza, así como los algoritmos de la etapa de recepción, se han realizado en FPGAs de la familia Spartan2 de Xilinx.

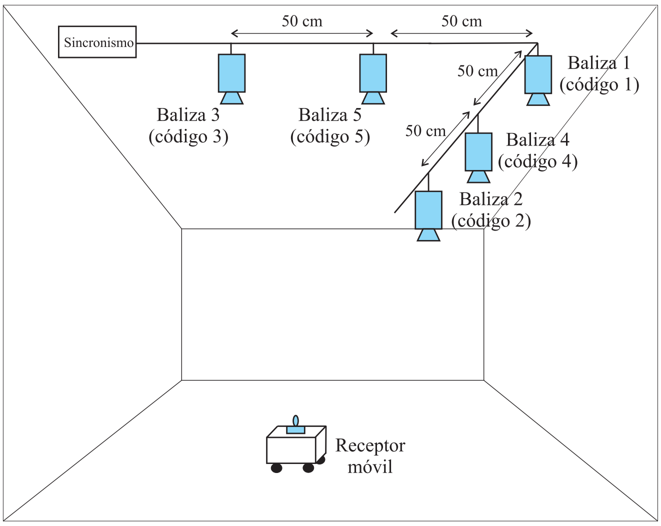


Figura 2.20: Representación del LPS propuesto por [UHJ+07].

Otro trabajo que merece la pena mencionar es el propuesto por Jiménez et al. en [JS05] para ambientes exteriores. Su objetivo es facilitar la localización de la posición de restos encontrados en excavaciones arqueológicas, tarea que conlleva un retardo importante en el avance de estas excavaciones. Según se muestra en la figura 2.21, el sistema consta de un conjunto de balizas situadas en una estructura elevada y que constituyen prácticamente toda la infraestructura a instalar, una baliza de referencia situada en la zona de medida, una serie de balizas inalámbricas que son las que el usuario maneja para conocer la posición de los restos y un sistema central conectado a un ordenador para la gestión del proceso de medida y cálculo de la posición tridimensional. En la versión mejorada de este sistema [PJG$^+$07], cada nodo puede actuar como emisor y como receptor, por lo que es posible medir tiempos de ida y vuelta. Se puede medir por tanto tiempos absolutos sin necesidad de una señal de sincronismo y calcular la posición a partir de la trilateración esférica (que es más precisa que la hiperbólica). Además, la emisión bidireccional compensa los efectos debidos a corrientes de aire mediante el promediado de los tiempos de ida y vuelta. Por otro lado, utiliza señales acústicas en vez de ultrasónicas para aprovechar el ancho de banda amplio de algunos altavoces y micrófonos, lo que permite una adecuada codificación de la señal emitida, a costa de admitir el sonido derivado de las emisiones. Se utilizan en este caso códigos Golay de 32 bits, modulados en QPSK con una portadora de 15 kHz con un ciclo por bit de código, que son muestreados en la recepción con una frecuencia de 150 kHz. Se comenten errores de posicionamiento menores de 1 cm con un nivel de confianza del 90 %.

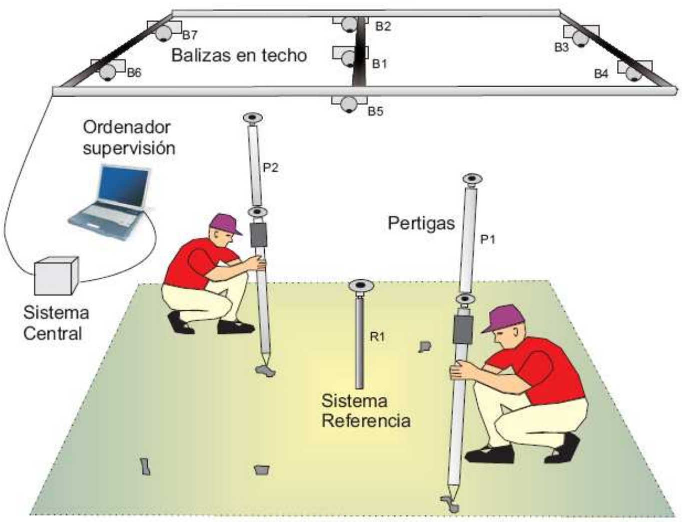


Figura 2.21: Esquema físico del sistema 3D-LOCUS [JS05].

En aquellas aplicaciones donde no es posible instalar una infraestructura externa de balizas, se pueden medir las distancias entre los móviles a posicionar en el entorno. Estas técnicas se denominan de *posición relativa* puesto que los móviles conocen su posición respecto a los demás, pero no su ubicación absoluta en el entorno. Para calcular esta posición relativa, los móviles tienen que interactuar comunicándose la información recolectada en cada uno de ellos. Cuando se requieren precisiones mayores y emisión simultánea se suelen codificar las emisiones, como ocurre en el trabajo propuesto por De Marziani et al.

[DMUH+07a] donde se utilizan conjuntos complementarios de secuencias emitidos mediante una modulación BPSK.

Códigos con zonas de correlación cero, como los LAS y T-ZCZ, están siendo usados con éxito en aplicaciones de comunicaciones [FFT07, YLHC07, LGZ07, FFTL08, RC08]. En esta tesis se estudiará la posible aportación de dichos códigos en la mejora de las prestaciones de sistemas sensoriales ultrasónicos.

Por último, conviene mencionar que, además de la codificación de la emisión, se encuentran otras propuestas para la medida precisa de tiempos de vuelo. Por ejemplo, en [TFB01] se propone combinar una medida poco precisa del tiempo de vuelo con un proceso de detección de fase de la señal recibida, obteniendo como resultado una medida mucho más fiable.

2.4. Otras áreas de aplicación de las secuencias binarias

Son muchas las áreas que requieren el uso de secuencias binarias con propiedades de correlación similares a las del ruido blanco gaussiano. En la sección anterior se ha visto, entre otros ejemplos, su aplicación en la evaluación no destructiva de materiales o en sistemas de localización. Sin embargo, además de aplicaciones sónar y radar basadas en medidas de TDV, existen otras que también requieren códigos con propiedades de auto-correlación o correlación cruzada específicas. En términos generales, aplicaciones que demandan buenas propiedades de auto-correlación son:

- *Identificación de parámetros en sistemas lineales.* Una de las primeras aplicaciones de estas secuencias fue la identificación de la respuesta al impulso de sistemas LTI (*Linear Time Invariant*) [BDG61][LDD94]. Si se introduce como entrada al sistema bajo test una secuencia pseudo-aleatoria o similar, la correlación cruzada entre la entrada del sistema y su salida es proporcional a su respuesta al impulso. Con este método, no resulta necesario alterar el funcionamiento normal del sistema para calcular su respuesta al impulso, por lo que ha sido el más utilizado para caracterizar sistemas lineales desde que se propuso en los años sesenta. Ejemplos de aplicaciones de la identificación de parámetros de un sistema LTI mediante medida de su respuesta al impulso, se encuentran en ingeniería de control, en comunicaciones para evaluar la respuesta del canal en tiempo real, radio-comunicaciones, etc.

- *Sincronización.* En la sincronización de bloque o de palabra, se establece una referencia temporal inicial que indica al sistema receptor el comienzo del bloque de datos. Para establecer la referencia temporal es habitual el uso de secuencias binarias con buenas propiedades de auto-correlación aperiódicas. Así, se transmite dicha secuencia binaria como un preámbulo antes de los datos; en el receptor se realiza la correlación con la señal recibida, identificando el pico de auto-correlación de la secuencia binaria como indicador del inicio de recepción del bloque de información. Como ejemplo, en [GD95] se han utilizado secuencias Barker como preámbulo de sincronización, y en [GM07] se aprovechan con el mismo fin las propiedades de los pares Golay.

- *Procesado bidimensional.* En arrays acoplados en fase, o en codificación simultánea en tiempo y frecuencia, también resulta necesario el uso de secuencias con un elevado pico de auto-correlación y lóbulos laterales bajos. Además, dadas las características de estos sistemas, deben utilizarse secuencias que permitan un funcionamiento multi-modo. En [LF80] se coteja el uso de secuencias Golay, frente a otros tipos, como las secuencias-m o Barker, en un sistema de transmisión formado por dos emisores/receptores y una serie de elementos de retardo que permiten el barrido de dos zonas distintas simultáneamente.

- *Criptografía.* Aunque las secuencias vistas hasta ahora se generan de forma determinista, no es fácil para un observador que pretenda decodificar dichas secuencias encontrar el patrón si no conoce el tipo de secuencias observadas. Los códigos Golay y, en general los CSS, resultan menos vulnerables, puesto que un observador que obtuviera un fragmento de dicha secuencia difícilmente podría reconstruir su totalidad dado que poseen máxima complejidad lineal. Otras secuencias, como las pseudo-aleatorias, tienen peores propiedades de encriptación, ya que pueden generarse a partir de registros de desplazamiento de longitud menor que el período de repetición de la secuencia. Por tanto, un observador que tuviera un fragmento suficiente de una secuencia pseudo-aleatoria podría reproducir la secuencia completa probando con distintas estructuras de registros de desplazamiento. Sin embargo, los códigos Golay sólo podrían generarse a partir de un registro de desplazamiento lineal igual a la longitud de las secuencias empleadas y con una realimentación única desde el inicio al final del registro. En cualquier caso, pueden utilizarse las secuencias pseudo-aleatorias como método de construcción de algoritmos de cifrado más complejos. Por ejemplo, dos secuencias pseudo-aleatorias pueden multiplexarse según el orden determinado por otra secuencia pseudo-aleatoria, o se puede generar una secuencia pseudo-aleatoria a partir de un generador con un reloj controlado por otra secuencia pseudo-aleatoria. Un trabajo interesante es el desarrollado en [KMM07], donde se evalúan las propiedades de secuencias pseudo-aleatorias y códigos ortogonales para su aplicación en un sistema de cifrado del habla.

Otras aplicaciones requieren el empleo de secuencias binarias con propiedades específicas de correlación cruzada, algunos ejemplos son:

- *Identificación de parámetros en sistemas lineales de múltiples entradas.* Para el caso de sistemas lineales con múltiples entradas y múltiples salidas (MIMO, *Multiple Input Multiple Output*), la técnica de medida de respuesta al impulso mediante correlación continúa siendo válida. En este caso, además de emplear secuencias con una función de auto-correlación impulsiva, deben tener una baja correlación cruzada entre ellas, de modo que interfieran lo menos posible unas medidas con otras. Un ejemplo es el sistema MIMO propuesto en [WA07] donde se utilizan códigos Golay polifásicos.

- *Sistemas de comunicaciones de acceso múltiple por división del código y ensanchado de espectro.* En los sistemas CDMA todos los usuarios transmiten con la misma frecuencia y de modo simultáneo en el tiempo. La separación entre usuarios se realiza asignando a cada uno de ellos un código distinto, de modo que los códigos asignados deben presentar correlaciones cruzadas lo más cercanas a cero posibles. Así, el número máximo de usuarios simultáneos en el sistema CDMA viene limitado por el número de secuencias con bajas correlaciones cruzadas entre ellas. El ensanchado de espectro hace referencia a que el código, o secuencia utilizada para modular los datos, extiende la energía de la señal a un rango de frecuencias mucho mayor que el necesario para transmitir la información de la señal. En el receptor se recupera la señal original utilizando una réplica sincronizada del código de ensanchado. En este ámbito, cuando se hace referencia a los elementos que componen la secuencia de datos se utiliza el término *bit*; y cuando se trata de los elementos del código de ensanchado se usa el término *chip*. Como en esta tesis los códigos de ensanchado de espectro no se usan para transmitir información, no se ha adoptado esta terminología, y se utiliza la palabra *bits* para denotar los elementos de las secuencias binarias de codificación bajo estudio. Son muchos los trabajos dedicados a la búsqueda de los códigos de ensanchado más adecuados en sistemas CDMA, algunos de ellos son: [DO99, Li03, Fan04, Woo02, DJ98]

- *Sistemas de ensanchado de espectro por salto en frecuencia.* Otra técnica de acceso a un canal de comunicación común es la de salto en frecuencia. En este caso, la secuencia de ensanchado alimenta un sintetizador de frecuencias, cuya salida se multiplica con la señal de datos modulada. De este modo, la señal resultante salta o cambia de frecuencia en el tiempo según la secuencia de ensanchado utilizada. Evidentemente, aparecerán interferencias cuando varios usuarios traten de acceder al mismo salto de frecuencia simultáneamente, por lo que deben elegirse códigos con correlaciones cruzadas bajas [FD96].

2.5. Implementación de los algoritmos de procesado de la señal ultrasónica

Si bien la técnica de compresión de pulsos supone una medida más precisa de los TDV, también implica el desarrollo de algoritmos de procesamiento más complejos, que conllevan una elevada carga computacional cuya implementación práctica puede llegar a superar los límites impuestos por la necesidad de trabajar en tiempo real, y que además pueden requerir plataformas de coste elevado. Por ello, los algoritmos y recursos necesarios para su implementación deben elegirse de modo que su complejidad sea mínima y sus tiempos de ejecución cumplan las restricciones derivadas del trabajo en tiempo real.

2.5.1. Esquemas de correlación eficiente

El método más óptimo para detectar una señal en un medio con ruido blanco gaussiano es el filtro acoplado, ya que maximiza la relación señal-ruido a la salida en el instante en que la forma de onda es recibida. En la práctica este filtro puede implementarse como un correlador que de manera continua realiza el sumatorio de los productos obtenidos tras multiplicar las L últimas muestras recibidas con las correspondientes L muestras de la secuencia patrón [Tur76].

En su modo más sencillo este filtro puede implementarse como un correlador paralelo según se muestra en la figura 2.22. La ventaja de esta implementación es que requiere un único ciclo de reloj para calcular una muestra de la función de correlación. Ahora bien, esta ventaja se torna en inconveniente cuando la longitud de la secuencia patrón es elevada, debido a la imposibilidad de hacer tal cantidad de multiplicaciones y sumas en un único ciclo de reloj sin recurrir a plataformas de alto coste. Una sencilla solución sería realizar una implementación secuencial, o serie, de dicho correlador (véase la figura 2.23), donde en cada ciclo de reloj una muestra del registro de entrada se suma o resta al resultado acumulado anterior en función del código patrón. Se necesitan L ciclos para obtener una nueva muestra de la función de correlación, y resetear el acumulador antes de repetir el proceso para un nuevo intervalo de datos de entrada. Evidentemente el principal problema de esta configuración es su lentitud. Muchos son los trabajos que tratan de buscar esquemas de correlación más eficientes, se destacan a continuación algunos de ellos.

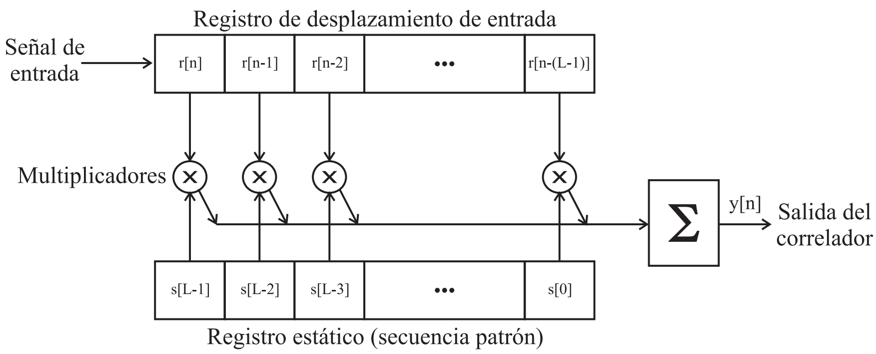


Figura 2.22: Diagrama de bloques de un correlador paralelo digital.

Povey y Grant [PG93] comparan los dos esquemas anteriores con uno híbrido serie-paralelo. La secuencia a detectar está formada por un grupo de M códigos pseudo-aleatorios de idéntico valor y longitud $L' = L/M$ que se concatenan con distintos desfases en función de otro código pseudo-aleatorio de menor longitud. La correlación con el primer código pseudo-aleatorio se realiza en paralelo, el resultado obtenido se corrige en fase con el segundo

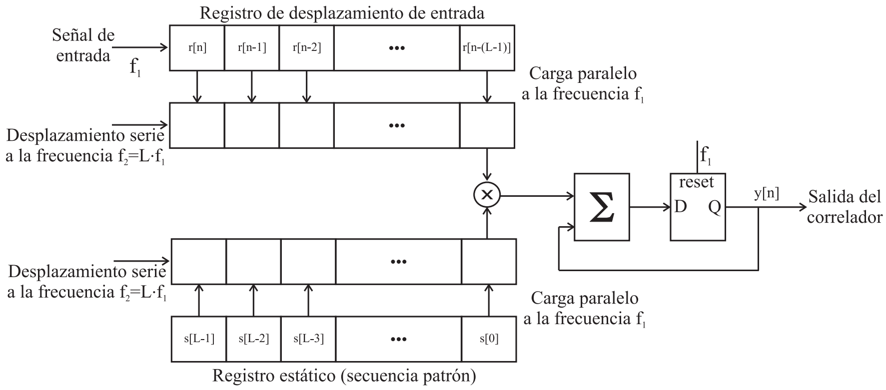


Figura 2.23: Diagrama de bloques de un correlador serie digital.

código pseudo-aleatorio y se acumula con el valor obtenido L' muestras antes. En este caso, se obtiene un nuevo resultado cada $L' = \frac{L}{M}$ ciclos de reloj.

En [LLK96] se propone un correlador diferencial que reduce a casi la mitad el número de multiplicaciones y sumas a realizar en la detección de secuencias pseudo-aleatorias (véase la figura 2.24). Si la salida del correlador en el instante $n-1$ es $y[n-1] = r[n-1]\cdot s[L-1] + r[n-2]\cdot s[L-2] + \cdots + r[n-L]\cdot s[0]$, y en el instante n es $y[n] = r[n]\cdot s[L-1] + r[n-1]\cdot s[L-2] + \cdots + r[n-(L-1)]\cdot s[0]$, entonces la ecuación en diferencias $y_d = y[n] - y[n-1]$ queda como:

$$\begin{aligned} y_d &= r[n]\cdot s'[L] + r[n-1]\cdot s'[L-1] + \cdots + r[n-(L-1)]\cdot s'[1] + r[n-L]\cdot s'[0] \\ &= r[n]\cdot s[L-1] + r[n-1]\cdot (s[L-2]-s[L-1]) + \cdots + r[n-(L-1)]\cdot (s[0]-s[1]) - r[n-L]\cdot s[0] \end{aligned} \tag{2.40}$$

Como los coeficientes $s[i]; 0 \leq i \leq L-1$ son 1 ó -1, los nuevos coeficientes obtenidos tras la ecuación en diferencias tienen valores $s'[L] = s[L-1]$, $s'[i] = s[i-1] - s[i] = \{-2, 0, 2\}; 1 \leq i \leq L-1$ y $s'[0] = -s[0]$. Para el caso de los códigos pseudo-aleatorios la mitad aproximadamente de estos coeficientes $s'[i]$ son ceros, reduciéndose en esa medida el número de multiplicaciones y sumas a realizar. Un año más tarde, Tan y Povey [TP97] estudian la posible implementación de este correlador en una FPGA de Altera, concluyendo que las operaciones de multiplicación por ± 1 se reducen a sumas y restas y aquellas que son por ± 2 pueden resolverse fácilmente con un desplazamiento. De este modo la correlación se implementa utilizando sólo sumadores, restadores y elementos de desplazamiento. Asimismo, muestran que los coeficientes $s'[L]$ y $s'[0]$ se pueden omitir para simplificar la función de correlación, asumiendo un pequeño error en los resultados obtenidos.

También para secuencias pseudo-aleatorias y, más específicamente para secuencias-m, Budisin [Bud89] plantea un correlador que reduce las operaciones a realizar basándose en la relación existente entre estas secuencias y las Walsh. El algoritmo se divide en tres pasos. En primer lugar se genera una matriz de secuencias donde cada columna representa un desfase distinto de una misma secuencia-m. Permutando las columnas de esta matriz en función del valor inicial de las N celdas de memoria del LFSR usado en la generación de las

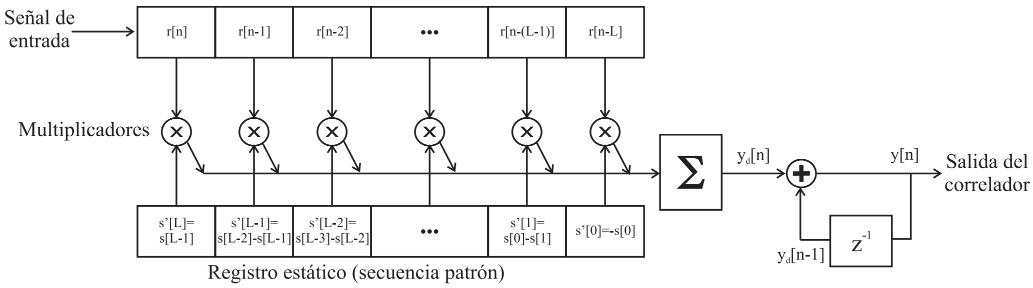


Figura 2.24: Diagrama del correlador diferencial propuesto por [LLK96].

secuencias-m, y añadiendo una primera columna y una última fila de ceros se consiguen las funciones Walsh buscadas, que aparecen desordenadas frente a las obtenidas con el algoritmo de J. J. Sylvester según puede verse en la figura 2.25. El segundo paso consiste en realizar la transformada rápida Walsh (FWT, *Fast Walsh Transform*) a las funciones Walsh obtenidas. Finalmente, se reordenan los resultados de la transformada Walsh considerando el orden en que las funciones Walsh aparecen en el algoritmo original -véase la expresión (2.11)-. Los resultados obtenidos corresponden a la correlación de la secuencia de entrada con todos los posibles desfases de una secuencia-m. Un correlador directo necesitaría realizar $L \cdot (L-1)$ operaciones, mientras que el propuesto por Budisin emplea $2 \cdot L \cdot log_2(L)$.

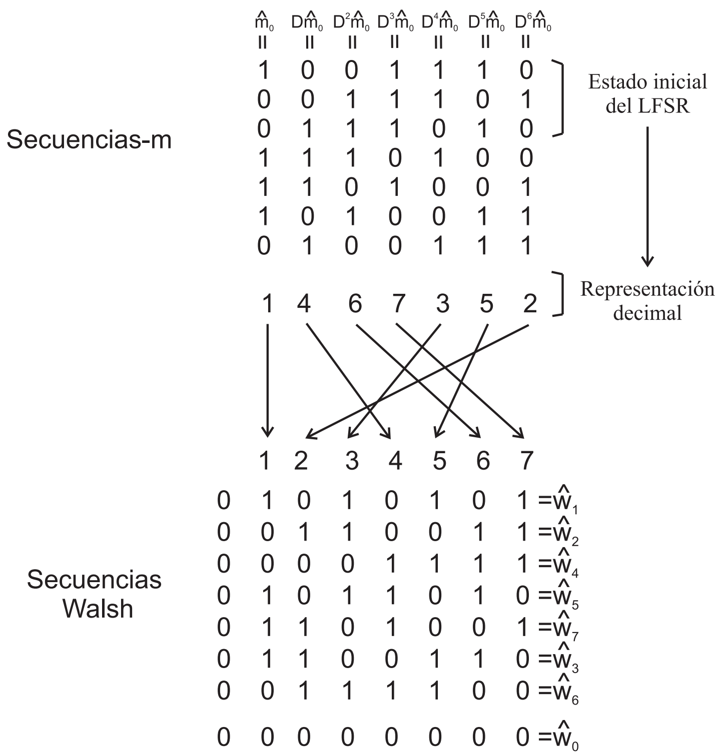


Figura 2.25: Obtención de códigos Walsh a partir de secuencias-m.

Popovic aprovecha la idea descrita en el párrafo anterior para proponer un correlador eficiente de secuencias Gold ortogonales [Pop97]. La generación de estas secuencias se realiza añadiendo un cero al final de dos secuencias-m y sumando en módulo-2 una de ellas (m'_1) con todos los posibles desfases de la otra (m'_2). Según se ha visto, los desfases de la secuencia m'_2 pueden agruparse en una matriz y a partir de ahí mediante permutación de columnas obtener las funciones Walsh asociadas. Además, si se trabaja en formato binario bipolar ($0 \rightarrow 1;\ 1 \rightarrow -1$) la suma módulo-2 se convierte en una multiplicación entre las columnas de dicha matriz y la otra secuencia m'_1. Según esto, el algoritmo de Popovic consta de tres etapas: multiplicación de la señal de entrada por la primera secuencia m'_1, reordenación según el algoritmo descrito en la figura 2.25 y realización de la FWT. El diagrama de bloques de la figura 2.26 resume el proceso. Nuevamente, el resultado corresponde a la correlación de la secuencia de entrada con todos los posibles desfases de la secuencia Gold ortogonal bajo estudio lo que supone un total de $2 \cdot L + L \cdot log_2 L$ operaciones[8]. La mejora frente a un correlador convencional es evidente.

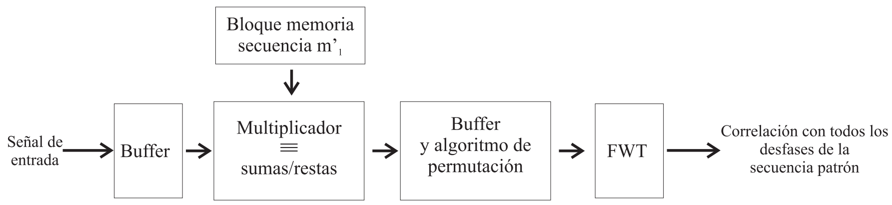


Figura 2.26: Diagrama de bloques del correlador eficiente de secuencias Gold ortogonales propuesto por [Pop97].

Otros autores como [JB98] realizan la detección de las secuencias pseudo-aleatorias mediante técnicas de correlación basadas en la transformada rápida de Fourier (FFT, *Fast Fourier Transform*). Sin embargo, aunque estas técnicas reducen el número de operaciones frente a un correlador directo, implican la realización de un considerable número de multiplicaciones, en contraste con el método directo en el que las multiplicaciones pueden reducirse a sumas y restas. Las multiplicaciones son más costosas computacionalmente que las sumas y restas, lo que incrementa la complejidad del circuito pudiendo resultar inviable la operación en tiempo real. Por otro lado, comparando este algoritmo con el propuesto por Budisin para secuencias-m, se necesitarían un total de $4 \cdot L \cdot log_2(L)$ multiplicaciones y $6 \cdot L \cdot log_2(L)$ sumas y restas frente a las $2 \cdot L \cdot log_2(L)$ sumas y restas que conlleva el algoritmo basado en la FWT.

Se adelantaba en la sección 2.2.4 de este capítulo la existencia de un correlador eficiente asociado a códigos Welti [Wel60], y se esbozaba la semejanza entre estos códigos y las parejas Golay. Específicamente para estas parejas Budisin [Bud91] y Popovic [Pop99] muestran un

[8] Los códigos ortogonales se utilizan para aplicaciones síncronas en las que el objetivo suele consistir en determinar la fase de la secuencia emitida. Para ello suele usarse un banco de correladores en paralelo proporcionando cada uno la correlación de la secuencia de entrada con un desfase concreto de la secuencia patrón.

método optimizado para la obtención de un correlador eficiente Golay (EGC). Este método permite simplificar el proceso de detección para el caso de trabajar con parejas Golay de longitud potencia de dos. El diagrama de bloques de la figura 2.27 muestra el algoritmo de forma esquematizada.

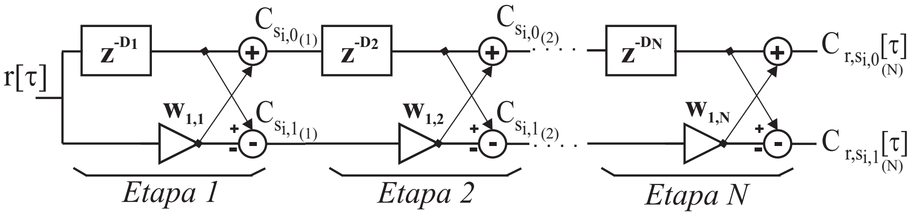


Figura 2.27: Diagrama de bloques del correlador eficiente Golay (EGC, *Efficient Golay Correlator*). Los valores D_n representan retardos y los coeficientes $w_{1,n}$ la semilla de generación del par Golay, pudiendo tomar los valores ± 1 [Bud91, Pop99].

A la salida del EGC se obtiene la correlación de la secuencia de entrada con las dos secuencias del par Golay. El número de operaciones se reduce desde $2 \cdot (L-1)$ sumas y $2 \cdot L$ multiplicaciones que requiere el correlador directo, a $2 \cdot log_2 L$ sumas y $log_2 L$ multiplicaciones utilizando el EGC. Al igual que ocurre en el correlador directo, las multiplicaciones pueden implementarse como sumas y restas dado el carácter binario de los coeficientes $w_{i,n}$. Por contra, el número de bits a almacenar en el EGC es mayor que en el correlador directo, aunque este número puede minimizarse escogiendo adecuadamente el orden de los retardos.

El EGC descrito anteriormente puede considerarse como un correlador eficiente de conjuntos de dos secuencias complementarias (2-ESSC, *Efficient Set of 2 Sequences Correlator*). Para el caso particular de conjuntos de cuatro secuencias complementarias F. Álvarez et. al [ÁUM+04] desarrollan también un algoritmo eficiente (4-ESSC), tal y como puede observarse en la figura 2.28. Para secuencias de longitud 4^N este correlador realiza solamente $8N$ operaciones, mientras que el correlador directo necesita $2 \cdot 4^{N+1} - 1$ (4^{N+1} productos y $4^{N+1} - 1$ sumas).

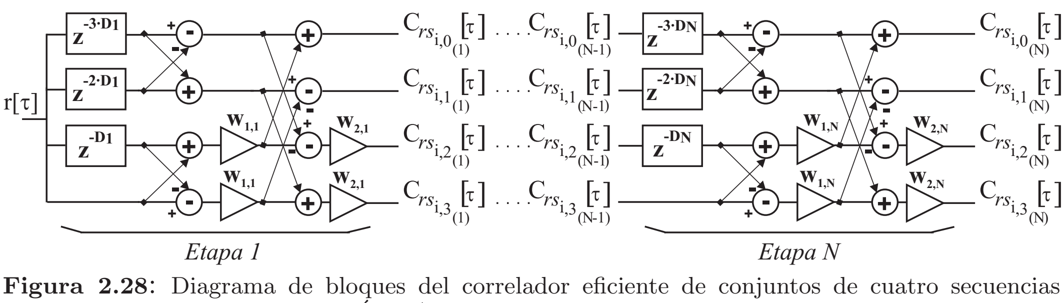


Figura 2.28: Diagrama de bloques del correlador eficiente de conjuntos de cuatro secuencias complementarias propuesto por [ÁUM+04].

C. De Marziani [DMUH+07b] amplia los trabajos anteriores proponiendo un esquema más general para la correlación eficiente de conjuntos de cualquier número $M = 2^m$ de secuencias complementarias de longitud $L = M^N$. La implementación de este correlador (en adelante M-ESSC) implica duplicar la estructura del correlador asociado a conjuntos de $M/2$ secuencias ($M/2$-ESSC); modificar el orden en el que aparecen los retardos de la estructura duplicada; multiplicar por el coeficiente $w_{1,n} \in \{-1, 1\}$ las salidas de la estructura inicial; añadir un conjunto de sumadores/restadores; y por último reordenar las salidas de cada etapa antes de conectarla a la siguiente, según puede observarse en 2.29. La tabla 2.5 muestra una comparativa entre las operaciones a realizar entre esta propuesta y un correlador directo.

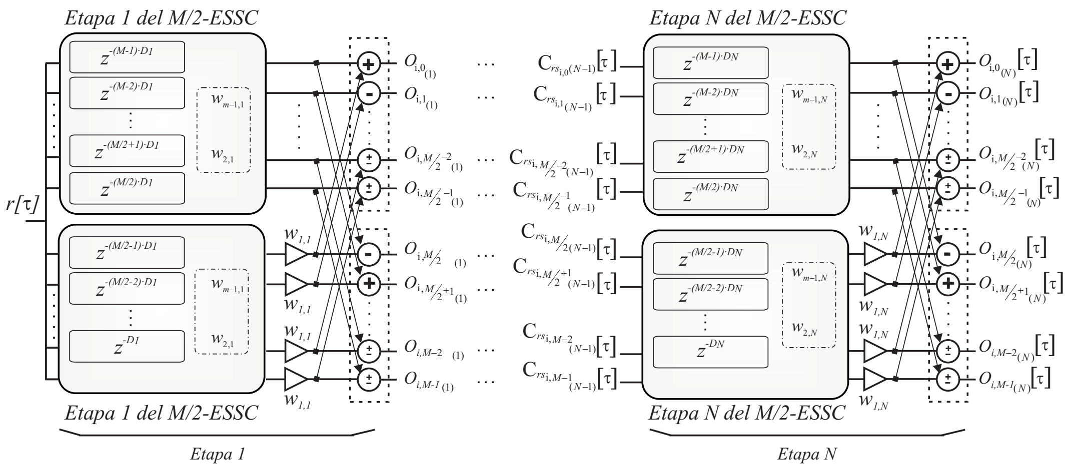


Figura 2.29: Diagrama de bloques del correlador eficiente de conjuntos de M secuencias complementarias propuesto por [DMUH+07b].

Implementación	Multiplicaciones	Sumas	Total operaciones
Directa	$M \cdot L = 2^m \cdot 2^{mN}$	$M \cdot (L-1) = 2^m \cdot (2^{mN} - 1)$	$2^m \cdot (2^{mN+1} - 1)$
M-ESSC	$2^{m-1} \cdot log_2 L = 2^{m-1} \cdot m \cdot N$	$2^m \cdot N \cdot m$	$3 \cdot 2^{m-1} \cdot m \cdot N$

Tabla 2.5: Comparativa del número de operaciones para la correlación de M-CSS mediante el método directo y la arquitectura propuesta por [DMUH+07b].

Según se ha comentado anteriormente, los códigos ortogonales generalizados (LA, LS, T-ZCZ, etc.) están cobrando cada vez más importancia en el sector de las comunicaciones. Por ello, uno de los objetivos de esta tesis se centra en la búsqueda de propuestas para la obtención de correladores eficientes asociados a dichos códigos.

2.5.2. Alternativas tecnológicas

Como se ha mencionado, los algoritmos de detección de las señales codificadas requieren una alta capacidad de cálculo, lo que conlleva una adecuada selección de la plataforma de implementación para poder trabajar en tiempo real. Existen numerosas alternativas, con consumos de potencia, costes, tiempos de diseño o velocidades de cómputo idóneas para determinadas aplicaciones e inadecuadas para otras. Se hace aquí una breve introducción a las arquitecturas más destacables, mostrando en el siguiente apartado algunos ejemplos de sistemas ultrasónicos que utilizan las empleadas en esta tesis.

Los *procesadores de propósito general* son plataformas de procesamiento genéricas, en las que con unos costes relativamente reducidos se puede abordar la implementación de prácticamente cualquier solución. Sus recursos y unidades operacionales se encuentran definidos al detalle, pero no así su funcionalidad, que admite una considerable flexibilidad y diversidad. De este modo, se puede variar fácilmente la función del sistema mediante instrucciones y sin alterar el hardware. Ahora bien, debido a su generalidad no optimiza el consumo de recursos y tiempos de operación, por lo que carecen de los rendimientos de otros sistemas diseñados a medida.

En los *sistemas específicos* como los ASIC (*Application Specific Integrated Circuit*) se pierde toda la generalidad anterior, para especializarse y adaptarse a la ejecución de un determinado proceso en una aplicación. Se dispone de un diseño totalmente a medida que mejora notablemente el rendimiento del proceso. El problema es que se trata de un diseño estático, incapaz de adaptarse a otras funciones, lo que redunda en un menor nicho de mercado y por tanto en unos costes más elevados, siendo poco rentables cuando el número de elementos necesarios es reducido. Además, el diseño y desarrollo de este tipo de circuitos requiere tiempos excesivos de validación y fabricación.

Otra alternativa es el uso de *procesadores digitales de señal* (DSPs). Se trata de microprocesadores específicamente diseñados para realizar el procesado digital de señales. Incluyen circuitería dedicada a ejecutar de forma rápida operaciones de multiplicación y acumulación, conocidas como MAC (*Multiply And Accumulate*); arquitecturas de memoria que permiten un acceso múltiple para cargar de forma simultánea varios operandos; modos especiales de direccionamiento; así como características de control de flujo de programa, diseñadas para acelerar la ejecución de operaciones repetitivas. Tienen la ventaja de poder programarse usando lenguajes de alto nivel, dada la existencia de múltiples compiladores con elevado grado de prestaciones. Sin embargo, el número de operaciones por ciclo de reloj es mucho menor que las conseguidas con dispositivos configurables tipo FPGA, siendo difícil en determinadas ocasiones el procesado en tiempo real.

Los *sistemas configurables* son una alternativa a los sistemas anteriores, cubriendo el vacío existente entre la flexibilidad de los procesadores genéricos y la especialización de los ASICs. Sus recursos internos pueden configurarse en el instante de desarrollo de la aplicación, de forma que poseen cierta flexibilidad. La capacidad de adaptación de estos sistemas permite modificar una determinada implementación ante nuevos cambios que hayan podido surgir en el entorno, o adaptarlo a ciertas variables como puede ser el ruido del sistema. Además, los recursos se emplazan allí donde son necesitados, de modo que desaparecen las confinaciones centralizadas de los procesadores genéricos, y con ellos los cuellos de botella de acceso a los mismos. Por otro lado, permiten una cierta especialización, adaptando el sistema al tamaño de los datos y mejorando el rendimiento del algoritmo, reduciendo el área y consumo de potencia necesarios. Disponen de un paralelismo de granularidad fina, lo que simplifica la distribución de instrucciones en el área de Silicio, y facilita el almacenamiento de datos en dispositivos externos. Asimismo, es posible configurar estos dispositivos dinámicamente, esto es, en tiempo de ejecución.

La figura 2.30 resume lo comentado anteriormente. A lo largo del eje horizontal se encuentran distribuidas las arquitecturas anteriores; el extremo izquierdo del eje implica una mayor flexibilidad, mientras que el derecho una mayor especialización. El eje vertical representa el nivel de esfuerzo en la programación del dispositivo.

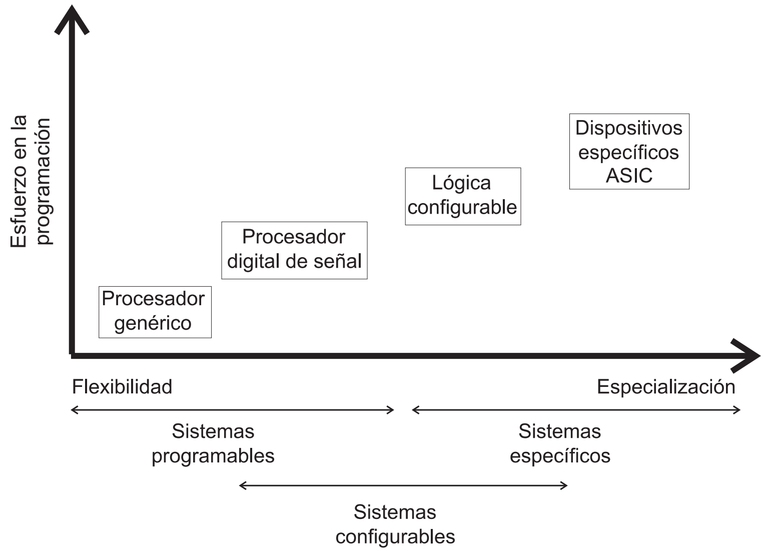


Figura 2.30: Clasificación de los sistemas de computación en función del grado de especialización y esfuerzo en la programación.

2.5.3. Tratamiento de la señal ultrasónica mediante sistemas configurables

Desde los primeros trabajos de procesamiento de la señal ultrasónica con correladores digitales se intuye la necesidad de definir el mejor soporte para la implementación de los algoritmos de procesado, aprovechando su naturaleza y buscando su ejecución en tiempo real, sin que ello suponga un coste elevado [HG88]. Frente a la propuestas realizadas con procesadores genéricos [KK95], redes de transputers [APKC92] o procesadores digitales de señal [JB98, HOB$^+$00] se sitúan los sistemas configurables. El uso de estas arquitecturas está especialmente indicado para llevar a cabo procesos con un alto nivel de paralelismo, con secuencias de datos de gran longitud, poca resolución en esos datos y con poco o ningún control sobre el proceso [MSH97], asemejándose mucho estas características a las de los algoritmos de proceso de señales sónar.

Las primeras propuestas de procesamiento de señales ultrasónicas sobre FPGAs se agrupan en torno al Departamento de Ingeniería Eléctrica y de Computación de la Universidad de Brigham Young (Utah). Así, en el trabajo propuesto por Graham y Nelson [GN98] se utiliza una red de FPGAs conectadas en anillo con una serie de bancos de memoria asociados a cada dispositivo, y un enlace hacia el exterior basado en un bus de comunicaciones genérico (véase la figura 2.31). El objetivo es el procesamiento de la señal de un sónar marino mediante la técnica de procesamiento de patrones (*beamforming*) en el dominio del tiempo. Se trata, en resumen, de sumar en fase las señales recibidas por un array de sensores considerando los retardos en la selección de muestras para conseguir que el sistema tenga un comportamiento direccional incluso cuando los sensores que lo forman sean omnidireccionales. La FPGA utilizada es una XC4085XL de Xilinx, cuyas prestaciones son contrastadas con un DSP SHARC ADSP2160 a 40 MHz de Analog Devices, concluyendo que, por cada FPGA serían necesarios un mínimo de 16 DSPs, siendo además la solución basada en FPGAs más económica.

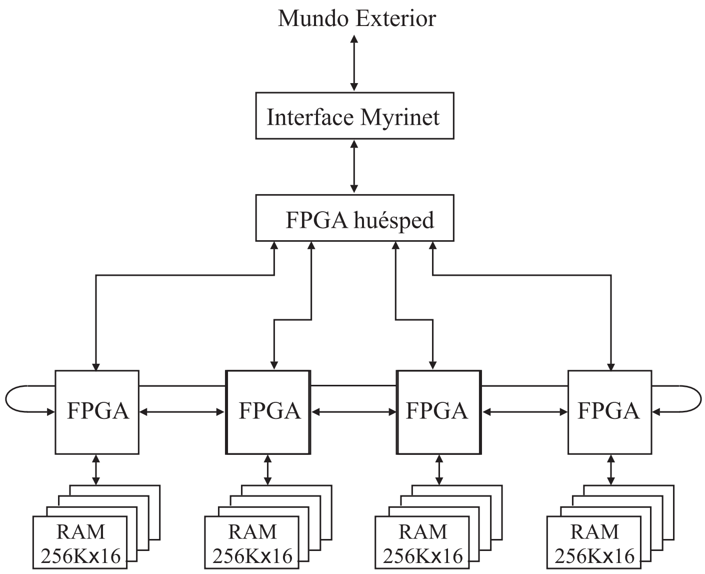


Figura 2.31: Arquitectura desarrollada por [GN98] para el procesamiento de la señal de un sónar marino.

En [Nel01] Nelson extiende el trabajo anterior para el caso de realizar el proceso de identificación de patrones en el dominio de la frecuencia, lo que implica un cálculo previo de las FFTs de las señales que llegan a cada sensor, confirmando que las FPGAs son realmente adecuadas para este tipo de aplicaciones de procesamiento de datos.

En [CQ04] se utiliza una FPGA Virtex-II de Xilinx para implementar un banco de filtros que modela el procesamiento de los ultrasonidos realizado en la cóclea de algunos murciélagos de gran tamaño. El modelo planteado recrea la traducción del movimiento de la membrana basilar en impulsos nerviosos, utilizando para ello un modelo neuronal de integración y disparo. Básicamente consiste en un banco de filtros paso banda, que representan la membrana basilar, seguido de una etapa de demodulación implementada con un rectificador de media onda y un filtro paso bajo. Se considera también un control automático de ganancia que imita el que realiza el murciélago en diferentes fases de su sistema de audición. La última etapa es un generador de impulsos, que consiste en aplicar a las señales resultantes del procesado anterior un umbral, de modo que si lo superan se genera un impulso. Para cada eco recibido, sólo se tiene en cuenta el primer impulso resultante de superar cada uno de los umbrales de los que consta la última etapa, siendo el resultado final el tiempo transcurrido desde que llega el estímulo hasta ese primer impulso. La figura 2.32 resume el proceso comentado anteriormente, mientras que la 2.33 muestra el diagrama de bloques de su implementación hardware. El sistema es capaz de procesar en tiempo real los cálculos correspondientes a 750 canales y 7500 impulsos, teniendo en cuenta que los ecos recibidos se muestrean a una frecuencia de 1 MS/s y con 12 bits de resolución, alcanzándose a medir tiempos de ida y vuelta de 65.5 ms, lo que equivale a una distancia aproximada de 11 m. Este trabajo constituye parte de un proyecto de colaboración entre distintas universidades europeas en donde se ha llevado a cabo la reproducción, a un nivel funcional, del sistema de eco-localización de los murciélagos; los resultados de este proyecto se resumen en [PLB$^+$05].

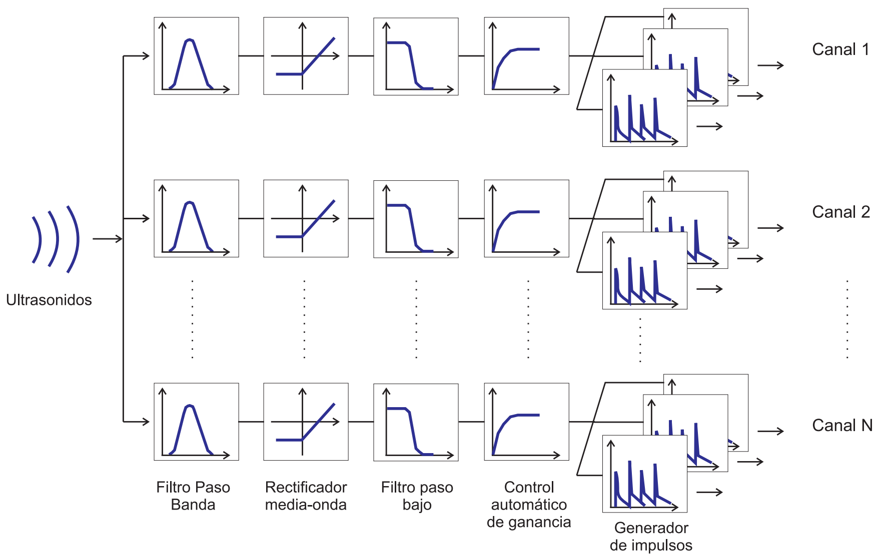


Figura 2.32: Diagrama de bloques del modelo de cóclea desarrollado en [PLB$^+$05].

Las aplicaciones ultrasónicas biomédicas, como la ecografía, también apuestan por las arquitecturas configurables para reducir el consumo de potencia, ruido y espacio que ocupan los sistemas tradicionales. Así, en [KSBH05] se propone la implementación de un SoC (*System-on-Chip*) sobre una FPGA XC2V200E de Xilinx para realizar el procesamiento digital de las señales ultrasónicas recibidas en un array de sensores. En la FPGA se lleva a cabo un proceso de formación de patrones, demodulación QPSK, filtrado, detección de envolvente y reajuste de amplitud de los valores obtenidos para obtener el grado de brillo

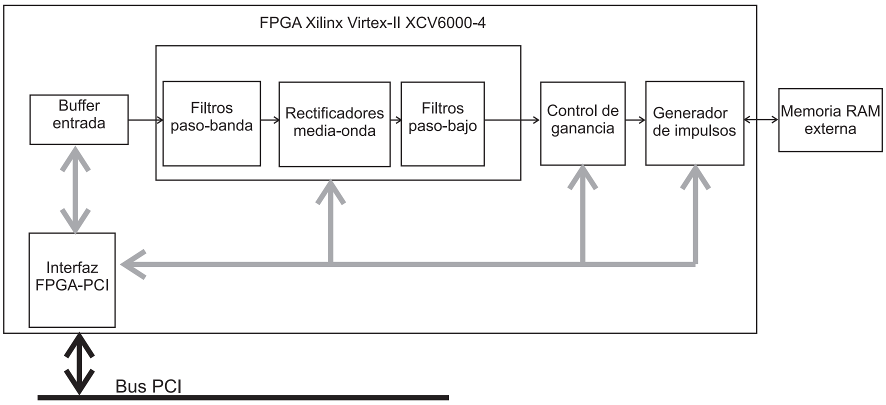


Figura 2.33: Estructura hardware de la arquitectura desarrollada en [CQ04, PLB+05] para el modelado de la cóclea de algunos murciélagos.

de cada pixel correspondiente al eco recibido, tal y como se muestra en la figura 2.34. El sistema es capaz de procesar 60 imágenes por segundo con una frecuencia de muestreo de 50 MHz, siendo el porcentaje de ocupación de ocupación de la FPGA (incluyendo circuitos de interfaz y buses) del 60 %.

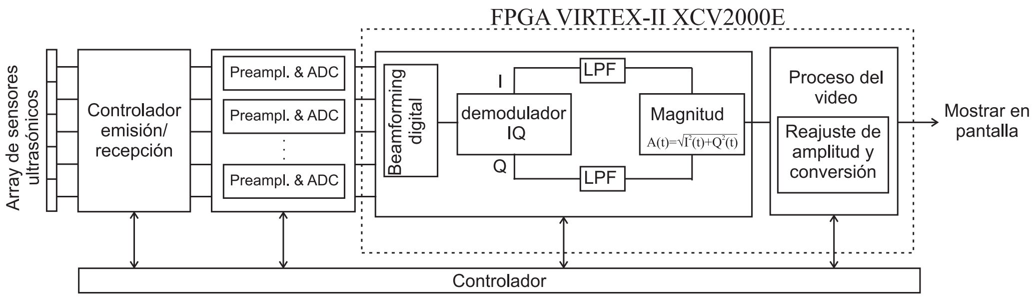


Figura 2.34: SoC basado en un array de sensores para procesamiento de imágenes ultrasónicas [KSBH05].

Por otro lado, en el Departamento de Electrónica de la Universidad de Alcalá se han empleado con éxito diversos dispositivos FPGA para el tratamiento de bajo nivel de la señal ultrasónica. En [UMG+99], por ejemplo, se ha utilizado una FPGA XC4005E de Xilinx para el procesamiento de ecos de la señal proporcionada por dos transductores que emiten simultáneamente un código Barker de 13 bits, uno a una frecuencia de 50 kHz y el otro a 56.25 kHz. En la FPGA se controlan los procesos de emisión y detección, de modo que en la emisión se genera la señal a emitir a la frecuencia especificada; y, en la detección, se realiza una primera correlación con el símbolo utilizado en la modulación, una segunda con el código emitido y se lleva a cabo la validación de picos. Este sistema permite la medida de tiempos de vuelo con una precisión de 2 μs para distancias menores de 3 m, y en tiempo real.

En un trabajo más reciente, A. Hernández [Á05] realiza una comparativa entre una arquitectura programable basada en el DSP de altas prestaciones C6701 de Texas Instruments y una arquitectura configurable basada en la FPGA XC4005E de Xilinx, para su uso en el procesamiento de bajo nivel de un sistema sónar que emplea una codificación basada en parejas Golay. Se demuestra en dicho trabajo que, si se desea operar en tiempo real, es más adecuada la arquitectura basada en la FPGA.

En [ÁHU+06] F. Álvarez presenta la implementación hardware en una única FPGA XCV1000E de Xilinx de todas las tareas implicadas en el proceso de detección de conjuntos de cuatro secuencias complementarias de longitud 64 (véase la figura 2.35). La ejecución de las tareas ha sido organizada en una estructura *pipeline*, de modo que todas ellas se ejecutan en un periodo de muestreo de valor 1.25 μs y con una latencia proporcional al tamaño de las ventanas de análisis usadas en el preamplificador digital y en el detector de picos.

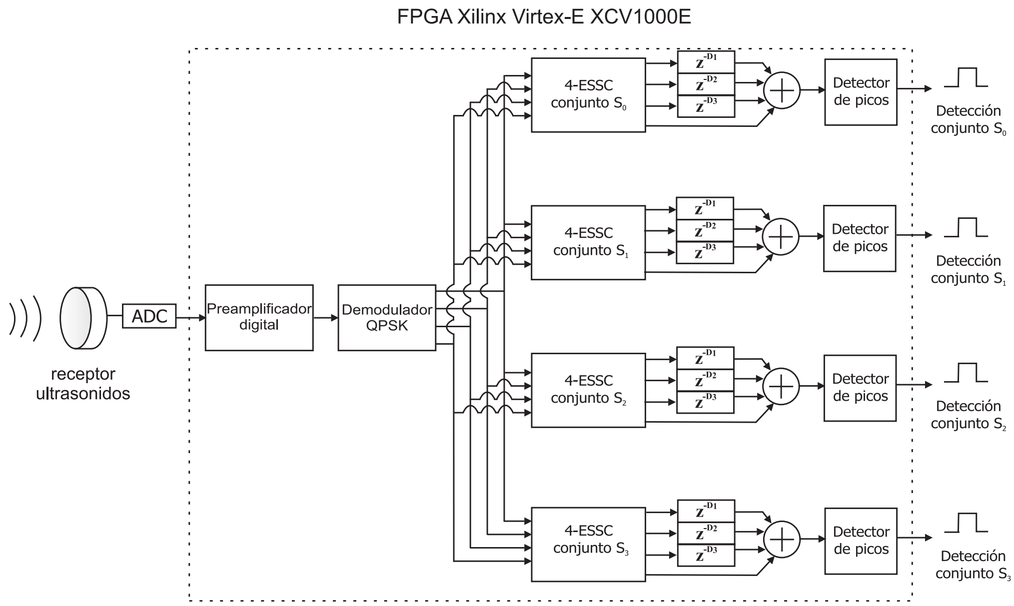


Figura 2.35: Diagrama de bloques del sistema de detección de conjuntos de cuatro secuencias complementarias propuesto en [ÁHU+06].

2.6. Estudio del interés científico de la codificación con secuencias a partir de las publicaciones asociadas

Existe un interés creciente en el estudio de secuencias con propiedades de correlación específicas. En las secciones anteriores ha quedado reflejado el amplio rango de áreas de aplicación que hacen uso de estas secuencias para la mejora de sus prestaciones. Es de destacar el intenso interés que en los últimos años ha aparecido como consecuencia del desarrollo de las comunicaciones móviles inalámbricas: los estándares actuales de tercera generación (3G) se basan en CDMA con códigos unitarios como los Kasami, Gold, Walsh-Hadamard, etc. que tienen una única secuencia por usuario; y los numerosos trabajos que están apareciendo anticipándose a la telefonía de cuarta generación proponen esquemas de codificación basados en conjuntos de secuencias complementarias y códigos ortogonales generalizados [Li03, CLY$^+$06, Che07, CCG08].

Por otro lado, un indicador del interés de la comunidad científica en la codificación con secuencias es la realización cada vez más frecuente de conferencias internacionales al respecto: "SETA, *Sequences and Their Applications*", "ISSSTA, *International Symposium on Spread Spectrum Techniques and Applications*", "ITW, *Information Theory Workshop*", "ISITA, *International Symposium on Information Theory and Applications*", "ISIT, *IEEE International Symposium on Information Theory*", "ICC, *IEEE International Conference on Communications*", etc. También en revistas especializadas existe un interés en publicar trabajos relacionados con técnicas de codificación y aplicaciones, así la revista "*IEEE Transactions on Information Theory*" desde 1998 cuenta con un editor asociado para artículos relacionados con secuencias.

Con objeto de constatar el esfuerzo investigador en el área de la codificación con secuencias binarias, se ha hecho una búsqueda en *google scholar*[9] [Sch08] del número de publicaciones total y por año asociadas a los esquemas de codificación usados en esta tesis: secuencias pseudo-aleatorias (PR), conjuntos de secuencias complementarias (CSS) y códigos ortogonales generalizados (GO). Los parámetros de búsqueda empleados se especifican al final de esta sección en la tabla 2.6.

En la figura 2.36 se representa el número de publicaciones indexadas en *google scholar* (a fecha de septiembre de 2008) correspondientes a cada uno de los esquemas de codificación evaluados. Obsérvese que las secuencias PR son las que acumulan un mayor número de publicaciones, de las cuales un 2.5 % están relacionadas con secuencias Kasami. La atención prestada a los CSS por la comunidad científica es también relevante; de estas publicaciones un 9.16 % corresponden a trabajos que emplean CSS para codificar emisiones acústicas. El menor número de publicaciones asociadas a códigos GO se justifica por su reciente aparición.

La figura 2.37 muestra, para cada codificación, el número de publicaciones acumulado desde el año 2000 hasta el 2008. En el año 2000 hubo un mayor número de publicaciones relacionadas con secuencias PR, sin embargo, desde entonces y hasta septiembre de 2008, el interés en CSS ha crecido notablemente, de modo que los CSS son los que han acumulado un mayor número de publicaciones en los últimos ocho años. En la figura 2.37.d se representa el porcentaje de publicaciones dedicado a cada codificación en función del año. Es de mencionar el aumento del número de publicaciones asociado a CSS y códigos GO experimentado en 2003. Este incremento se debe en gran medida a la introducción de CDMA en la telefonía móvil y los esfuerzos investigadores realizados desde la fecha en solventar las limitaciones

[9] *Google scholar* [Sch08] es una herramienta de búsqueda que indexa documentación técnica, y ofrece resultados dirigidos a la comunidad científica, entre los que se incluyen trabajos de investigación, artículos, informes técnicos, libros y tesis.

de los códigos CDMA tradicionales [CF02, Che07].

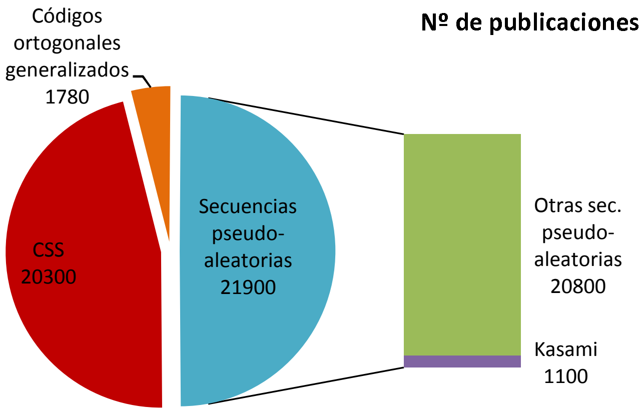
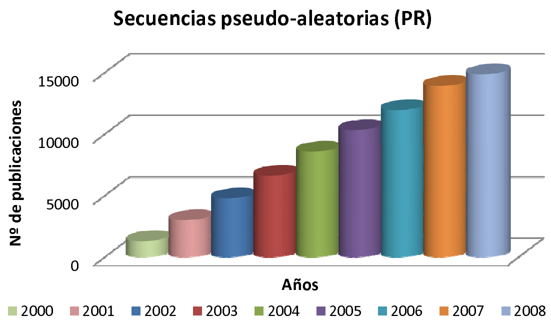
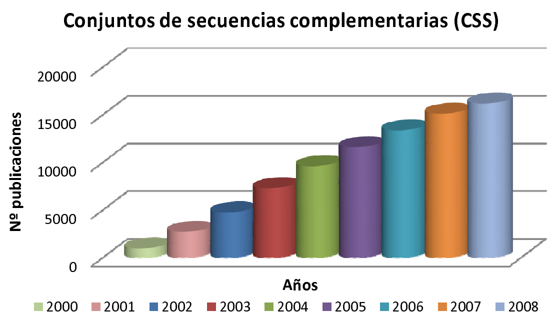


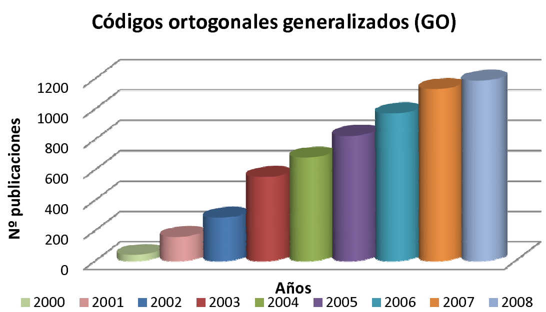
Figura 2.36: Distribución del número de publicaciones en función del código empleado.



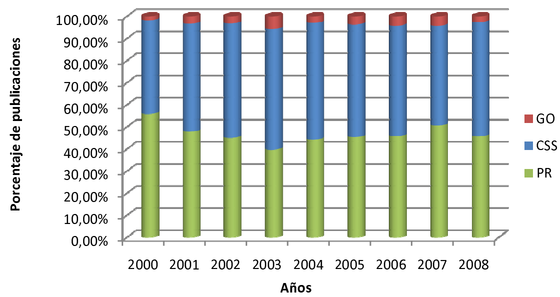
(a)



(b)



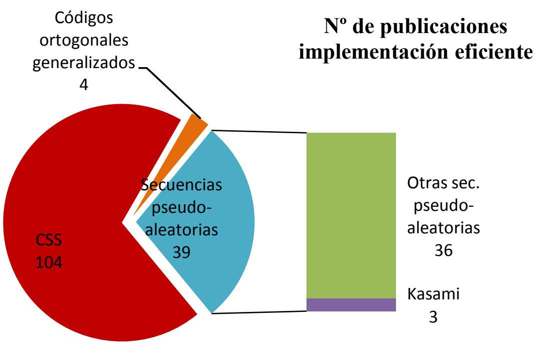
(c)



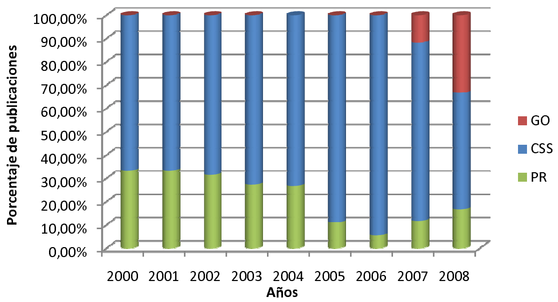
(d)

Figura 2.37: a), b) y c) Número de publicaciones acumulado por año en función del esquema de codificación; d) Porcentaje de publicaciones correspondiente a cada codificación en función del año.

Finalmente, en la figura 2.38 se hace una revisión del número de trabajos que proponen o emplean correladores eficientes asociados a algún tipo de codificación concreta. En este sentido, destacan los CSS con un mayor número de publicaciones, debido al extendido uso del correlador eficiente Golay [Bud91, Pop99] y la más reciente propuesta de correladores eficientes de CSS [ÁUM+04, DMUH+07b]. En cuanto a correladores de códigos GO, tres de las cuatro publicaciones son trabajos derivados de esta tesis [PUH+07a, PUH+07b, PUH+08]. La publicación restante propone códigos polifásicos con zonas de correlación cero en caso de emisión periódica, y realiza la detección de los mismos utilizando la transformada rápida de Fourier (FFT) [TZW08].



(a)



(b)

Figura 2.38: a) Distribución del número de publicaciones sobre propuestas de correlación eficiente asociadas a cada esquema de codificación; b) Porcentaje de publicaciones con propuestas de codificación eficiente asociadas a una codificación concreta en función del año.

2.7. Objetivos planteados

Las medidas ultrasónicas en el aire son bastante sensibles a errores que influyen negativamente en la precisión de estos sistemas. La influencia del ambiente, la potencia de emisión, o el tipo de reflexión producida implican un error variable en la determinación de los tiempos de vuelo. En aplicaciones que requieren de una cierta precisión se suele procesar la señal ultrasónica empleando técnicas de compresión de pulsos y filtrado óptimo parecidas a las usadas en radar o sónar marino, en donde se emite una señal codificada cuyo eco es buscado y localizado mediante correlación. Según se ha mostrado en la sección 2.2, existe una gran variedad de códigos empleados en la codificación de estas señales, y dependiendo de las características de la aplicación concreta, deben elegirse unos u otros. Sin embargo, estas técnicas suponen un aumento importante de la complejidad computacional del procesamiento de la señal ultrasónica, abriendo un campo de investigación para el desarrollo de algoritmos eficientes que minimicen su carga computacional de cara a operar en tiempo real.

Publicaciones que proponen o emplean los esquemas de codificación binarios estudiados en esta tesis.	
PR	"pseudo-random sequence" OR "pseudo-random code" OR "pseudo-random codes" OR "pseudo-random sequences" OR "pseudorandom code" OR "pseudorandom codes" OR "pseudorandom sequence" OR "pseudorandom sequences" OR "Gold codes" OR "Kasami codes" OR "maximal-length sequences" OR "Gold sequences" OR "Kasami sequences"
Kasami	"Kasami codes" OR "Kasami code" OR "Kasami sequences" OR "Kasami sequence"
CSS	"golay codes" OR "golay sequences" OR "complementary sequences" OR "complementary codes" OR "complementary set of sequences"
GO	"LS codes" OR "LS code" OR "zero correlation zone" OR "zero correlation zones" OR "interference free window" OR "LAS-CDMA" OR "Ear ZCZ" OR "Three ZCZ" OR "zero correlation window" OR "ZCZ codes" OR "ZCZ sequences"
Publicaciones que emplean CSS en aplicaciones sensoriales acústicas.	
CSS	"golay codes" OR "golay sequences" OR "complementary sequences" OR "complementary codes" OR "complementary set of sequences" AND "ultrasounds" OR "ultrasonic" OR "acoustic" OR "acoustical" OR "sonar"
Publicaciones que proponen o emplean correladores eficientes asociados a los esquemas de codificación binarios estudiados en esta tesis.	
CSS	"pseudo-random sequence" OR "pseudo-random codes" OR "pseudo-random sequences" OR "pseudorandom sequence" OR "pseudorandom sequences" OR "Gold codes" OR "Kasami sequences" OR "maximal-length sequences" AND "efficient matched filter" OR "efficient correlator" OR "efficient pulse compressor" OR "efficient despreaders" OR "fast correlator"
Kasami	"Kasami codes" OR "Kasami code" OR "Kasami sequences" OR "Kasami sequence" AND "efficient matched filter" OR "efficient correlator" OR "efficient pulse compressor" OR "efficient despreaders" OR "fast correlator" OR "simplified matched filter" OR "low-cost matched filter"
CSS	"golay codes" OR "golay sequences" OR "complementary sequences" OR "complementary codes" OR "complementary set of sequences" AND "efficient matched filter" OR "efficient correlator" OR "efficient pulse compressor" OR "efficient despreaders" OR "fast correlator" OR "golay correlator" OR "efficient set of sequences correlator"
GO	"LS code" OR "zero correlation zone" OR "zero correlation zones" OR "interference free window" OR "LAS-CDMA" OR "zero correlation window" AND "efficient matched filter" OR "efficient correlator" OR "efficient pulse compressor" OR "efficient despreaders" OR "fast correlator" OR "efficient LS correlator"

Tabla 2.6: Parámetros de búsqueda de publicaciones utilizado en el *Google Scholar* [Sch08].

Teniendo en cuenta estas consideraciones, el objetivo global de esta tesis es analizar la eficacia de distintos tipos de secuencias binarias en la mejora de las prestaciones de un sistema de sensores ultrasónicos, en donde tienen lugar múltiples emisiones simultáneas, aperiódicas y con detección asíncrona o cuasi-síncrona. Este objetivo global conlleva una serie de objetivos parciales que se detallan a continuación:

- *Búsqueda de sub-conjuntos óptimos de códigos para su aplicación en un sistema sensorial ultrasónico asíncrono y estudio comparativo de cotas de correlación.*

  Toda la algoritmia de alto nivel que se utiliza en aplicaciones de navegación, sistemas de posicionamiento local o sistemas de clasificación y detección de obstáculos requiere la disponibilidad de tiempos de vuelo medidos con precisión suficiente. Por ello los códigos empleados en la codificación deben tener un pico de auto-correlación claramente identificable y lóbulos laterales bajos, de modo que incluso en ambientes ruidosos sea posible la detección de la señal. Además, estas aplicaciones requieren medidas simultáneas desde distintos puntos para controlar en la mayor medida posible el entorno. Deben considerarse, por tanto, códigos con bajas o nulas correlaciones cruzadas entre ellos de modo que transductores cercanos no se interfieran. La primera fase de esta tesis consiste en analizar los códigos que mejor se adaptan al sistema ultrasónico descrito anteriormente en cuanto a sus valores de correlación. Para ello, en primer lugar y para cada familia de códigos, se llevará a cabo una búsqueda exhaustiva para determinar en función del número de usuarios simultáneos los códigos concretos con menores lóbulos laterales en la ACF y CCF. Posteriormente se realizará un análisis comparativo que tenga en cuenta los siguientes parámetros: valor del pico principal de correlación en relación a la longitud de la secuencia, cota de auto-correlación, número de códigos incorrelados o pseudo-incorrelados disponibles, cota de correlación-cruzada y capacidad de reducir la distancia mínima que puede medirse con un único transductor.

- *Propuesta de un nuevo esquema de generación de pares T-ZCZ.*

  El uso de pares T-ZCZ en aplicaciones cuasi-síncronas resulta de gran interés debido a las propiedades de correlación aperiódica de los mismos: presentan zonas de correlación cero sin que ello suponga una pérdida de la ganancia de proceso y permiten un mayor número de emisiones simultáneas sin interferencias que los pares Golay (siempre y cuando la diferencia máxima entre los TDV sea inferior a la ZCZ alrededor del origen).

  En esta línea la propuesta llevada a cabo consiste en el desarrollo de un nuevo esquema de generación de códigos T-ZCZ que mejore las prestaciones de los ya existentes en términos de correlación: las zonas de interferencias deben estar más acotadas y los valores de cota máxima deben ser inferiores. Además, el proceso de generación y correlación de estos nuevos pares debe poder realizarse de modo eficiente, esto es, empleando un número de operaciones significativamente menor que las requeridas por una implementación directa.

- *Desarrollo de algoritmos óptimos de proceso.*

  En la detección, cada nueva muestra debe ser tratada al ritmo en que éstas son suministradas por el sistema de adquisición para poder obtener resultados en tiempo real. En este sentido, otro objetivo de la tesis implica el desarrollo de nuevos algoritmos de generación y correlación eficientes de códigos ortogonales generalizados (LS y T-ZCZ) que permitan reducir al máximo el número de operaciones a realizar en el proceso de detección y que redunden en una fácil implementación hardware. Además se adaptarán algoritmos ya existentes, como el propuesto en [DMUH+07b], para favorecer su implementación en arquitecturas configurables. Las necesidades computacionales de cada una de las propuestas realizadas se compararán con las requeridas por un correlador directo serie tradicional.

- *Implementación de estos algoritmos en una arquitectura hardware de computación que permita su ejecución en tiempo real.*

  Una vez determinadas las bondades de los distintos tipos de códigos, se implementarán en hardware los algoritmos de detección de aquellos con mejores prestaciones. La plataforma escogida será una arquitectura configurable puesto que, según ha quedado demostrado en trabajos previos [Her03], este tipo de plataformas se adapta perfectamente a las tareas de procesamiento de bajo nivel de los sistemas sónar. Se realizarán diseños genéricos que puedan adaptarse a las condiciones cambiantes que pueda presentar el entorno, y se tratará de reducir al máximo los recursos empleados y los tiempos de ejecución.

- *Realización de pruebas simuladas y experimentales en un LPS ultrasónico.*

  Finalmente, se llevará a cabo un conjunto de simulaciones y pruebas experimentales en un LPS ultrasónico desarrollado ex profeso con objeto de comprobar que el comportamiento de los diferentes códigos se ajusta a lo esperado. Se evaluarán aspectos tales como la capacidad de discriminar varias emisiones simultáneas, la inmunidad al ruido, la resistencia al efecto cerca-lejos y la respuesta de los diversos códigos ante el multi-camino.

Capítulo 3

Búsqueda de códigos óptimos para detección asíncrona

Una mayor precisión en la medida de TDV implica el uso de técnicas de codificación y proceso de señal parecidas a las usadas en radar. Como ya se ha visto en la sección 2.2, hay una gran variedad de códigos disponibles, cuya elección está subordinada a la aplicación a desarrollar.

Muchas de estas aplicaciones, como los sistemas de posicionamiento local o de clasificación de reflectores, requieren esquemas de codificación que permitan tener varias emisiones simultáneas sin interferencia entre ellas. En estos casos, códigos como los Barker o las secuencias-m quedan descartados porque no disponen de un número suficiente de códigos con correlaciones cruzadas bajas.

Por otro lado, en aquellos casos donde se trabaja con elementos móviles es habitual usar sistemas de detección asíncrona, ya que los ecos pueden llegar en cualquier momento dependiendo de las posiciones de transductores y reflectores. Como resulta obvio, no tiene sentido en estas aplicaciones el uso de códigos ortogonales, tipo Walsh, que requieren un sincronismo estricto. Los códigos ortogonales generalizados, o cuasi-ortogonales generalizados, podrán utilizarse siempre que los distintos ecos se reciban dentro de la ventana libre de interferencias. En sistemas síncronos esto se traduce en usar una ventana de tamaño mayor al tiempo de dispersión del canal; y en asíncronos implica ubicar los emisores lo suficientemente cerca unos de otros y delimitar la zona de posicionamiento de los receptores móviles, de modo que la llegada de los TDV no exceda el tamaño de dicha ventana.

En algunos sistemas la emisión de la señal ultrasónica codificada no se realiza de forma continua o periódica, sino que las emisiones se espacian en el tiempo. Además del consiguiente ahorro energético, se consigue que ecos de una misma emisión recibidos en distintos instantes de llegada, por propagación multicamino, no se confundan con ecos directos de otra emisión consecutiva. Se necesitan en estos casos códigos con buenas propiedades de correlación aperiódica, como los CSS, LAS o T-ZCZ descritos en las secciones 2.2.4 a 2.2.7.

Las secuencias pseudo-aleatorias se consideran normalmente para aplicaciones periódicas, estando sus propiedades de correlación definidas para ese caso (véase la sección 2.2.3). Es posible, no obstante, utilizar estas secuencias en entornos aperiódicos; aunque los lóbulos laterales de auto-correlación y los valores de correlación cruzada obtenidos no se ajustan a los valores perfectamente acotados del caso periódico. Esto es, mientras que en el caso periódico las cotas de correlación obtenidas son siempre las mismas para una determinada longitud de secuencia, en la emisión aperiódica difieren en función del LFSR utilizado en la generación de la secuencia-m original y de la fase de dicha secuencia-m. Así, puede ocurrir que las cotas obtenidas sean bastante peores que las del caso periódico, o que por contra lleguen a mejorarlas. Resulta por tanto de gran utilidad un estudio que determine las secuencias pseudo-aleatorias más óptimas para trabajar en entornos aperiódicos. En [SP80] se apuntaba ya esta idea, y se muestra una lista de LFSRs que proporcionan códigos Gold de 31 bits con lóbulos laterales reducidos en su ACF aperiódica. El handicap de este estudio reside en que el número de cálculos a realizar para encontrar las secuencias pseudo-aleatorias más óptimas se dispara cuando la longitud de las secuencias aumenta (para un conjunto de M secuencias de longitud L se obtiene un total de L^M nuevos conjuntos combinando los posibles desfases de las M secuencias originales. Además, para cada combinación de desfases, deben evaluarse $\frac{1}{2}M(M-1)$ funciones de correlación cruzada). En este capítulo se lleva a cabo una búsqueda de aquellas secuencias Kasami que menores cotas de correlación presentan para el caso de emisión aperiódica. La elección de secuencias Kasami frente a otras secuencias pseudo-aleatorias se justifica por sus menores valores de correlación cruzada.

Otro aspecto a considerar son las restricciones derivadas de los transductores utilizados y su circuitería de excitación asociada. Por un lado es necesario adaptar los códigos a emitir a la banda de frecuencias del transductor; y por otro, cuando el código está formado por varias secuencias (como sucede en los CSS y parejas T-ZCZ) es necesario recurrir a esquemas de transmisión que permitan la emisión eficiente de todas las secuencias en el menor tiempo posible. En el caso particular de parejas Golay (2-CSS) y T-ZCZ puede emplearse una modulación digital QPSK para realizar la transmisión simultánea de las dos secuencias que forman el código [Her03, HUG$^+$04]. Cuando el número de secuencias de un CSS es más elevado, el método más simple y que menores necesidades de ancho de banda presenta consiste en ordenar las secuencias, mediante concatenación o entrelazado de los bits que las componen, y emitirlas empleando una modulación BPSK [ÁUM$^+$06, Á05]. Sin embargo, esto implica una degradación de las propiedades ideales de los CSS [DMUH$^+$06]. En este capítulo se dedica una sección a la búsqueda de los CSS que minimicen las interferencias debidas al mecanismo de generación utilizado.

Existe además un límite inferior, conocido como *zona ciega*, para las distancias que pueden ser medidas con un único transductor. El acoplamiento de la señal emitida con la etapa de recepción impide que cualquier objeto situado a una distancia tal que el eco reflejado llegue al transductor antes de que se haya emitido todo el pulso ultrasónico sea detectado correctamente [Mas99]. La reducción de esta zona ha sido estudiada en [HUM+06] donde se demuestra que, dependiendo de las propiedades de auto-correlación de las señales emitidas, puede reducirse en un gran porcentaje dicho límite inferior permitiendo la detección de reflectores próximos al transductor. En estos casos interesan secuencias que permitan distinguir con claridad el pico principal de los lóbulos laterales cuando se reciba un porcentaje pequeño del eco reflejado.

Resumiendo, en este capítulo se lleva a cabo una selección de las secuencias con menores cotas de correlación (incluso en el caso de pérdida de bits al inicio de la secuencia) de cara a una aplicación en donde van a tener lugar múltiples emisiones simultáneas, aperiódicas y con detección asíncrona. Aunque en este trabajo la aplicación está enfocada al uso de sistemas sensoriales ultrasónicos, son múltiples los campos que requieren secuencias con propiedades similares [LZL05, CLY+06, KMM07]. En cualquier caso, teniendo en cuenta las consideraciones anteriores, ha quedado demostrado en los párrafos previos que los códigos de mayor interés, y por tanto los que se han analizado en este capítulo, son: Kasami, CSS (incluyendo en este grupo los códigos Golay), LS y T-ZCZ.

A modo de recordatorio se presenta en la tabla 3.1 un resumen de las propiedades principales de los códigos detallados en la sección 2.2.

3.1. Cotas para correlación aperiódica

Como ya se adelantaba en la sección 2.2, un método bastante utilizado para determinar los atributos de secuencias binarias es calcular su cota de correlación θ, que para el caso de emisiones aperiódicas se define como:

$$\begin{aligned} \theta &= \max\{\theta_{AC}, \theta_{CC}\} \\ \theta_{AC} &= \max\left\{\frac{|C_{a_m,a_m}[\tau]|}{C_{a_m,a_m}[0]}; \forall m \in [0,\cdots,M-1]; \forall \tau \neq 0\right\} \\ \theta_{CC} &= \max\left\{\frac{|C_{a_m,a_s}[\tau]|}{C_{a_m,a_m}[0]}; \forall m,s \in [0,\cdots,M-1]; m \neq s; \forall \tau\right\} \end{aligned} \tag{3.1}$$

En donde C_{a_m,a_s} denota la correlación aperiódica entre dos secuencias binarias a_m y a_s; y M es el número de secuencias de la familia, esto es, el número de secuencias incorreladas o pseudo-incorreladas disponibles. En el caso de CSS y T-ZCZ el valor de cota θ se calcula sumando las funciones de correlación aperiódica de cada una de las secuencias del código.

Secuencia	Requerimientos sincronismo	Ideadas para emisión periódica	$N, m, n \in \mathbb{N}$	Longitud máxima	Nº secuencias del código ($\star$)	Nº códigos incorrelados	ACF en $\tau = 0$	ACF máx en $\tau \neq 0$	CCF máx.
Walsh-Hadamard	$\tau = 0$	No	$\forall N$	2^N	1	2^N	2^N	—	—
Barker	$\forall \tau$	No	—	13	1	1	13	-1	—
secuencias-m preferidas	$\forall \tau$	Si	$\forall N$	$2^N - 1$	1	($\star\star$)	$2^N - 1$	-1	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
Gold	$\forall \tau$	Si	$\forall N$	$2^N - 1$	1	$2^N + 1$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
conjunto pequeño Kasami	$\forall \tau$	Si	$\mathrm{mod}(N, 2) = 0$	$2^N - 1$	1	$2^{\frac{N}{2}}$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N}{2} \rfloor} \rvert$
conjunto grande Kasami	$\forall \tau$	Si	$\mathrm{mod}(N, 2) = 0$	$2^N - 1$	1	$2^{\frac{N}{2}} (2^N + 1)$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
CSS	$\forall \tau$	No	$\forall\ N, m$	2^{mN}	2^m	2^m	2^{mN}	0	0
LS ($\star\star\star$)	$\lvert\tau\rvert \leq W$	No	$\forall\ N, m$	$2^{m(2+N)} + (2^m - 1)W$	1	2^{2m}	$2^{m(2+N)}$	0, en $1 \leq \lvert\tau\rvert \leq W$	0, en $0 \leq \lvert\tau\rvert \leq W$
T-ZCZ método 1	$\lvert\tau\rvert \leq W$	No	$\forall\ m, n$	2^{2n+m+1}	2	2^{n+1}	2^{2n+m+2}	0, en $1 \leq \lvert\tau\rvert \leq W$	0, en $0 \leq \lvert\tau\rvert \leq W$
T-ZCZ método 2	$\lvert\tau\rvert \leq W$	No	$\forall\ m, n$	2^{2n+m}	2	2^{n+1}	2^{2n+m+1}	0, en $1 \leq \lvert\tau\rvert \leq W$	0, en $0 \leq \lvert\tau\rvert \leq W$
T-ZCZ método 3	$\lvert\tau\rvert \leq W$	No	$\forall\ m, n$	2^{2n+m-1}	2	2^{n+1}	2^{2n+m}	0, en $1 \leq \lvert\tau\rvert \leq W$	0, en $0 \leq \lvert\tau\rvert \leq W$

$\star$ Cuando el número de secuencias del código es mayor que la unidad, los resultados de correlación mostrados en las tres últimas columnas se refieren a la suma de correlaciones de las secuencias que componen dicho código.

$\star\star$ El número de secuencias-m pseudo-incorreladas depende de la longitud escogida, siendo insuficiente en la mayoría de los casos (véase la Tabla 2.2).

$\star\star\star$ Secuencias LS generadas según [ZYH05].

Tabla 3.1: Resumen de las propiedades de correlación de los códigos binarios más empleados en la compresión de pulsos.

Existen trabajos previos que permiten obtener cotas mínimas teóricas asociadas a determinado tipo de códigos. Por ejemplo, en la emisión aperiódica de una familia compuesta por M códigos binarios $\in \{-1, 1\}$ de longitud L, con una única secuencia por código ($K = 1$), resulta interesante calcular las siguientes cotas:

Cota de Welch [Wel74]:

$$\theta^2 \geq \frac{M-1}{2ML-M-1} \tag{3.2}$$

Cota de Sarwate [Sar79]:

$$\frac{2(L-1)}{(M-1)} \cdot \theta_{AC}^2 + (2L-1) \cdot \theta_{CC}^2 \geq 1 \tag{3.3}$$

Cota de Levenshtein [Lev99]:

$$\theta^2 \geq \frac{3MLN - 3L^2 - MN^2 + M}{3L^2(MN-1)}, \quad 1 \leq N \leq L \tag{3.4}$$

La cota de Levenshtein depende de un valor entero N, que si se elige convenientemente da lugar a las siguientes simplificaciones:

$$\begin{aligned} \theta^2 &\geq \frac{1}{L} - \frac{2}{L\sqrt{3M}}, \quad \textit{cuando } M \geq 3 \\ \theta^2 &\geq \frac{1}{L} - \frac{1}{L^2}\left\lceil \frac{\pi \cdot L}{\sqrt{8M}} \right\rceil, \quad \textit{cuando } M \geq 5 \end{aligned} \tag{3.5}$$

De (3.2) y (3.5) se deduce que, para un número de códigos elevado (especialmente cuando M crece más rápido que L), la cota mínima proporcionada por Levenshtein es mayor que la de Welch, y por tanto más restrictiva. Por otro lado, la cota de Sarwate (3.3) establece una relación de compromiso entre los valores de auto-correlación y correlación cruzada aperiódicos.

En el caso de CSS existen M conjuntos incorrelados entre sí, cada uno con M secuencias complementarias. En consecuencia, la cota teórica correspondiente a la suma de las funciones de correlación de las secuencias de dichos conjuntos es cero. Welch [Wel74] propone una nueva cota mínima que contempla las interferencias que aparecen en las funciones de correlación aperiódicas cuando más de M conjuntos se emiten simultáneamente. Si $\mu > M$ es el número de CSS a emitir, M es el número de secuencias de cada conjunto y L la longitud de las mismas, la cota de Welch para *correlación aperiódica multi-canal* es:

$$\theta^2 \geq \frac{\mu/M - 1}{2\mu L - \mu - 1} \tag{3.6}$$

Los códigos ortogonales generalizados (LA [Li99], LS [TSG00, ZYH05], T-ZCZ [ZLH04, LGZ07, FFTL08]) y cuasi-ortogonales generalizados [TF01b] son de aparición más reciente que los códigos de compresión de pulsos clásicos (ortogonales [Syl67], Barker [Bar53], pseudo-aleatorias [Gol67b, Gol67a, Kas68] y CSS [Gol61, TL72]), y las cotas tradicionales, como las de Welch, Sarwate o Levenshtein, no contemplaban su existencia. Así, varios trabajos han encaminado sus esfuerzos a la búsqueda de cotas mínimas que puedan servir de referencia a la hora de evaluar estos nuevos esquemas de codificación.

Considerando códigos cuasi-ortogonales generalizados, cuyas características de correlación se resumen en (2.10), las siguientes cotas permiten calcular el valor mínimo de correlación aperiódica teórico en la zona de reducidas interferencias (LCZ, $0 \leq |\tau| \leq W_q$, representando W_q una zona alrededor del origen donde las interferencias son de muy bajo valor):

Cota de Tang-Fan [TF01a]. Para secuencias ternarias con valores $\in \{-1, 0, 1\}$.

$$\theta^2 \geq \frac{(M-1)W_q - L + 1}{(MW_q - 1)(L + W_q - 1)} \tag{3.7}$$

Cota de Peng-Fan [PF03, PF04]. Proporciona resultados más ajustados que la cota de Tang-Fan cuando se trata de secuencias binarias con valores $\in \{-1, 1\}$.

$$3NL^2\theta_{AC}^2 + 3L^2(N+1)(M-1)\theta_{CC}^2 \geq 3ML - 3L^2 + 3MNL - 2MN - MN^2, \quad 0 \leq N \leq W_q \tag{3.8}$$

Si $\theta = \theta_{AC} = \theta_{CC}$ la cota de Peng-Fan queda:

$$\begin{aligned} \theta^2 &\geq \tfrac{3ML - 3L^2 + 3MLN - 2MN - MN^2}{3L^2(MN + M - 1)}, && 0 \leq N \leq W_q \\ \theta^2 &\geq \tfrac{\sqrt{3M} - 2}{\sqrt{3M}L - 1}, && W_q \geq \sqrt{3/M}L - 1 \end{aligned} \tag{3.9}$$

Cuando las interferencias dentro de la LCZ son nulas, las secuencias cuasi-ortogonales generalizadas (GQO) se convierten en ortogonales generalizadas (GO) y esa zona pasa a denominarse zona de correlación cero (ZCZ). Por otro lado, cuando $W_q = 0$ y $\epsilon = 0$ las secuencias ortogonales generalizadas pasan a ser secuencias ortogonales; y si $W_q = 0$ se obtienen secuencias binarias tradicionales. De este modo, las secuencias GQO pueden considerarse como un caso general que incluye a las secuencias GO, a secuencias ortogonales y a binarias tradicionales como casos especiales. Es más, las cotas de Welch, Sarwate o Levenshtein pueden derivarse de las cotas mínimas teóricas obtenidas para los códigos GQO.

Las cotas mínimas dadas por las expresiones (3.2) a (3.9) sirven como referencia para evaluar el comportamiento de distintos tipos de códigos. En concreto, proporcionan el valor mínimo de interferencia que podría conseguirse utilizando una familia cualquiera de M códigos de longitud L. Esto no quiere decir que sea posible alcanzar esos valores mínimos teóricos con todos los códigos de compresión de pulsos disponibles. Por ejemplo, las secuencias Kasami periódicas presentan un comportamiento más óptimo respecto a la cota de Welch que las secuencias Gold periódicas. Además, debe tenerse en cuenta que algunas de estas cotas presentan valores más realistas que otras; de hecho ya se ha visto que para secuencias tradicionales la cota de Levenshtein es la más adecuada y para GQO es la de Peng-Fan.

En las siguientes secciones de este capítulo se realiza un análisis comparativo de las cotas de correlación obtenidas con códigos Kasami, CSS, LS y T-ZCZ para el caso de emisión aperiódica. Además, para cada uno de estos códigos se ha llevado a cabo una búsqueda exhaustiva con objeto de encontrar las combinaciones de μ emisores simultáneos que menores valores de cota $\theta = \max(\theta_{AC}, \theta_{CC})$ presentan. En esta búsqueda se ha tratado de ponderar por igual la cota de auto-correlación (θ_{AC}) y la de correlación cruzada (θ_{CC}). Así, para un número de emisores μ simultáneos y una longitud L de secuencia concreta, se calculan aquellas combinaciones que menores valores $f_\theta = 0.5 \cdot \theta_{CC} + 0.5 \cdot \theta_{AC}$ presentan. Posteriormente, se eligen de entre esas combinaciones las que tienen valores θ_{AC} y θ_{CC} más parecidos, de modo que para un mismo valor de f_θ se consideran sólo las de menor cota θ. Una vez obtenidos dichos sub-conjuntos de μ secuencias, ya óptimos desde el punto de vista de la cota de correlación, se establece como criterio adicional, para poder tomar una decisión, priorizar la cota de correlación cruzada seleccionando los grupos de menor θ_{CC}. Los resultados obtenidos cuando se prioriza la auto-correlación son muy similares sino iguales. Por último, los valores de cota resultantes de la búsqueda son confrontados con los teóricos para poder establecer una comparación.

Cabe señalar también que, en el estudio de las propiedades de auto-correlación aperiódica de una familia de códigos, además de la cota de auto-correlación (θ_{AC}) existe otro criterio denominado Factor de Mérito (MF, Merit Factor) que proporciona una medida de todos los lóbulos laterales de la ACF frente el pico principal [Gol77]:

$$MF = \frac{C_{a_m,a_m}(0)^2}{2\sum_{\tau=1}^{L-1} |C_{a_m,a_m}(\tau)|^2} \tag{3.10}$$

3.2. Secuencias Kasami

3.2.1. Selección de secuencias Kasami con buenas propiedades de correlación aperiódica

En el caso de emisión aperiódica las cotas de correlación obtenidas varían en función de la fase de la secuencia emitida. Para un conjunto de M secuencias de longitud L implica evaluar un total de $M \cdot L$ auto-correlaciones, y $\frac{L^M}{2}(M^2 - M)$ correlaciones cruzadas. Una búsqueda exhaustiva de todas las posibles combinaciones resulta prohibitiva cuando el número de secuencias M aumenta, por ello suelen utilizarse procedimientos de búsqueda subóptimos. Estos procedimientos pueden agruparse en tres según el método utilizado para la selección de secuencias con cotas de correlación aperiódica bajas: por un lado están los que limitan el algoritmo de búsqueda a un número menor de posibilidades; por otro, los que relajan las condiciones de búsqueda de modo que secuencias cuya cota está por debajo de un determinado umbral son validadas como óptimas; y finalmente, algunos procedimientos combinan los dos métodos anteriores [Kri03].

En este trabajo se ha realizado una selección de aquellas secuencias Kasami que menores cotas de correlación presentan atendiendo al primer procedimiento. Así, se ha llevado a cabo una búsqueda exhaustiva considerando los distintos conjuntos de secuencias Kasami que pueden formarse cambiando la realimentación y valor inicial del LFSR utilizado en la generación de la secuencia-m de la cual derivan.

En la tabla A.1 del apéndice A se muestran las distintas opciones de realimentación que dan lugar a secuencias-m. Eligiendo LFSRs con un número N par de celdas de memoria se obtienen secuencias-m de longitud $L = 2^N - 1$, a partir de las que se construyen los conjuntos Kasami correspondientes. Además, para un LFSR dado la variación de su estado inicial supone obtener el mismo conjunto de secuencias Kasami, pero con las secuencias que lo componen ordenadas de distinta forma y todas desfasadas un mismo valor. Como ejemplo, en (3.11) se muestra un conjunto de secuencias Kasami generado a partir de una secuencia-m m_1 obtenida con un LFSR con realimentación y valor inicial cualesquiera. Utilizando el mismo LFSR, pero cambiando su valor inicial, se obtiene la misma secuencia-m pero con un desfase distinto a la anterior ($m'_1 = D^x m_1$); con esto, el nuevo conjunto de secuencias Kasami es el indicado en (3.12). En emisión aperiódica, se consideran distintos los L conjuntos Kasami obtenidos a partir de los posibles desfases de una misma secuencia-m. Como consecuencia, el número de correlaciones cruzadas a realizar es $N_{LFSR} \cdot \frac{L}{2}(M^2 - M)$; en donde N_{LFSR} es el número de polinomios primitivos, o realimentaciones, posibles de LFSRs con número N par de celdas que dan lugar a secuencias de longitud máxima $L = 2^N - 1$; y $M = 2^{\frac{N}{2}}$ el número de secuencias del conjunto Kasami.

$$Kasami = \{m_1,\ m_1 \oplus m_2,\ m_1 \oplus Dm_2,\ m_1 \oplus D^2 m_2, \cdots,\ m_1 \oplus D^{L'-1} m_2\} \tag{3.11}$$

$$\begin{aligned} Kasami' &= \{m'_1,\ m'_1 \oplus m'_2,\ m'_1 \oplus Dm'_2,\ m'_1 \oplus D^2 m'_2, \cdots,\ m'_1 \oplus D^{L'-1} m'_2\} \\ &= \{D^x m_1,\ D^x(m_1 \oplus D^2 m_2),\ D^x(m_1 \oplus m_2),\ D^x(m_1 \oplus D^{L'-1} m_2), \cdots,\ D^x(m_1 \otimes D^3 m_2)\} \end{aligned} \tag{3.12}$$

Las cotas mínimas de correlación obtenidas para grupos de μ secuencias Kasami simultáneas de longitud L se resumen en la tabla 3.2. La realimentación del LFSR que da lugar a dichas cotas θ se ha representado mediante los coeficientes $h_0 h_1 h_2 \cdots h_N$ que acompañan al polinomio primitivo $h(x) = h_0 \cdot x^N + h_1 \cdot x^{N-1} + \cdots + h_{N-1} \cdot x + h_N$. Por otro lado, el valor inicial del LFSR $\hat{a}_0[N]\hat{a}_0[N-1] \cdots \hat{a}_0[0]$ se ha mostrado en formato decimal según $\sum_{i=0}^{N-1} 2^i \cdot \hat{a}_0[i]$. Las μ secuencias concretas del conjunto de M secuencias a las que corresponden esas cotas θ se han especificado atendiendo al orden en que aparecen en (3.11); de modo que la secuencia 0 corresponde a m_1, la secuencia 1 a $m_1 \oplus m_2$, la 2 a $m_1 \oplus Dm_2$, y así sucesivamente. Finalmente, y con objeto de poder establecer una comparación, se ha incluido la cota de Levenshtein $\theta_{Leven.}$ asociada a los valores de M y L de los conjuntos Kasami evaluados; así como la cota θ_{PER} que se hubiese obtenido en caso de emisión periódica.

θ	θ_{AC}	θ_{CC}	$h(x)$	valor inicial LFSR	secuencias	$\theta_{Leven.}$	θ_{PER}
Secuencias Kasami L=15 y M=4							
$\mu = 2$							
0.3333	0.2667	0.3333	11001	11	0, 3	–	0.3333
$\mu = 3$							
0.3333	0.3333	0.3333	11001	11	0, 1, 2	–	0.3333
$\mu = 4$							
0.3333	0.3333	0.3333	11001	11	0, 1, 2, 3	0.1679	0.3333
Secuencias Kasami L=63 y M=8							
$\mu = 2$							
0.1746	0.1587	0.1746	1011011	16	0, 3	–	0.1428
$\mu = 3$							
0.1905	0.1746	0.1905	1110011	10	0, 5, 6	–	0.1428
$\mu = 4$							
0.2063	0.1905	0.2063	1100001	50	0, 1, 2, 4	–	0.1428
$\mu = 5$							
0.2063	0.1905	0.2063	1100001	55	0, 1, 3, 5, 7	–	0.1428
$\mu = 6$							
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5	–	0.1428
$\mu = 7$							
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5, 6	–	0.1428
$\mu = 8$							
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5, 6, 7	0.0982	0.1428
Secuencias Kasami L=255 y M=16							
$\mu = 2$							
0.0980	0.0902	0.0980	100101101	124	0, 13	–	0.0666
$\mu = 3$							
0.1059	0.0941	0.1059	100101101	111	5, 7, 9	–	0.0666
$\mu = 4$							
0.1059	0.1020	0.1059	100101101	55	0, 5, 7, 15	–	0.0666
$\mu = 5$							
0.1137	0.0980	0.1137	100101101	169	0, 5, 7, 8, 10	–	0.0666
$\mu = 6$							
0.1137	0.0980	0.1137	100101101	115	0, 5, 7, 8, 10, 14	–	0.0666
$\mu = 7$							
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6, 7, 8	–	0.0666
$\mu = 8$							
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6 7, 8, 10	–	0.0666
$\mu = 9$							

Continuación de la Tabla 3.2

θ	θ_{AC}	θ_{CC}	$h(x)$	valor inicial LFSR	secuencias	$\theta_{Leven.}$	θ_{PER}
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6 7, 8, 10, 13	–	0.0666
$\mu = 10$							
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6, 7, 8, 10, 13, 14	–	0.0666
$\mu = 11$							
0.1137	0.1059	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13	–	0.0666
$\mu = 12$							
0.1137	0.1059	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 9, 11, 13, 14	–	0.0666
$\mu = 13$							
0.1137	0.1098	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14	–	0.0666
$\mu = 14$							
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13	–	0.0666
$\mu = 15$							
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14	–	0.0666
$\mu = 16$							
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	0.0532	0.0666

Tabla 3.2: Valores mínimos de cota aperiódica para secuencias Kasami, en función de su longitud L y número μ de usuarios simultáneos.

La figura 3.1 representa gráficamente los datos de la tabla 3.2. Puede observarse como, para una determinada longitud de secuencia, aumentar el número de usuarios simultáneos hasta el número de secuencias disponibles en el conjunto no afecta apenas a la cota aperiódica. Por otro lado, y como era de esperar, longitudes mayores de secuencias implican menores valores de cota θ. Comparando los valores de cota de correlación aperiódica con los obtenidos en caso de emisión periódica se tiene que, excepto para secuencias de longitud $L = 15$, en el resto los valores de cota son peores en emisión aperiódica. Finalmente, es fácil comprobar que, conforme aumenta la longitud de las secuencias, la cota de correlación periódica se acerca asintóticamente a la cota de Levenshtein, mientras que la aperiódica es aproximadamente el doble.

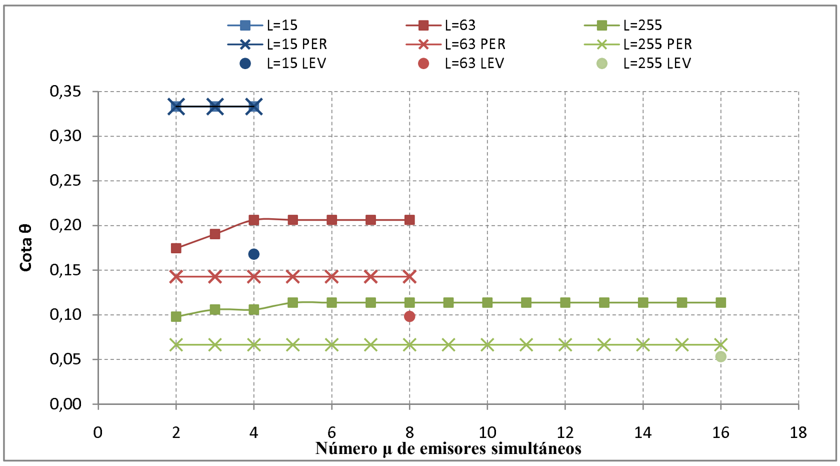


Figura 3.1: Cotas de correlación mínimas para secuencias Kasami, en función de su longitud L y del número μ de usuarios simultáneos.

3.2.2. Influencia de la recepción parcial de la secuencia Kasami en la cota de auto-correlación

En la medida de tiempos de vuelo es habitual utilizar un mismo transductor tanto en emisión como en recepción, lo que supone la aparición de una *zona ciega* en las cercanías del transductor debido al acople de la emisión al circuito de recepción. Esto implica que no es posible recibir los ecos correctamente, bien porque el circuito de recepción esté saturado o bien porque se haya deshabilitado [SB01, Kle04, HUG$^+$04].

Cuando se utilizan técnicas de compresión de pulsos, aumentar la longitud de la secuencia binaria supone una mayor inmunidad frente al ruido, pero también tiempos de emisión mayores, lo que se traduce en un incremento del tamaño de la zona ciega. En estos casos resulta de utilidad analizar cómo se comporta la función de auto-correlación frente a la pérdida bits en el inicio de la secuencia [HUM$^+$06]. Básicamente, esta pérdida lleva asociada una reducción del pico principal de auto-correlación y un aumento de los lóbulos laterales (cuanto mayor sea la pérdida, mayor serán los lóbulos laterales en comparación con el principal), lo que puede derivar en una medida incorrecta de los TDV.

Las figuras 3.2 y 3.3 ilustran el comportamiento de los códigos Kasami cuando se pierde el inicio de la secuencia debido, por ejemplo, a un reflector dentro de la zona ciega. En concreto, la figura 3.2 muestra los valores de cota de auto-correlación (θ_{AC}) aperiódica obtenidos en función del porcentaje de bits perdidos al inicio de la secuencia, para distintas longitudes de código y cuatro emisores simultáneos. Para cada longitud, los cuatro emisores elegidos son aquellos de menor cota θ, cuyos valores pueden consultarse en la tabla 3.2. Se puede observar en esta figura como los mejores resultados de θ_{AC} corresponden a las secuencias de mayor longitud, pudiéndose conseguir tamaños de zona ciega similares a los obtenidos con secuencias más cortas. Por ejemplo, considerando una zona ciega equivalente a la recepción de 51 bits: para secuencias de longitud $L = 63$ supone la recepción de aproximadamente un 80 % del eco reflejado y una cota de auto-correlación $\theta_{AC} = 0.3$; mientras que para secuencias de longitud $L = 1023$ equivale a la recepción de únicamente el 0.05 % del eco, siendo la cota $\theta_{AC} = 0.49$ y por tanto suficiente para discriminar el pico principal de los lóbulos laterales.

En la figura 3.3 se representa el decremento en la magnitud del pico principal según aumenta el porcentaje de bits sin recibir, así como los valores máximos de los lóbulos laterales de auto-correlación. Los resultados obtenidos para un número de usuarios $\mu \neq 4$ son similares a los mostrados en las figuras 3.2 y 3.3.

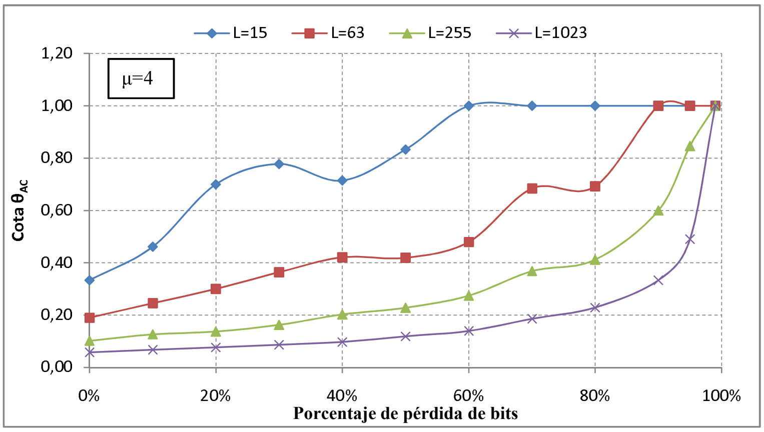
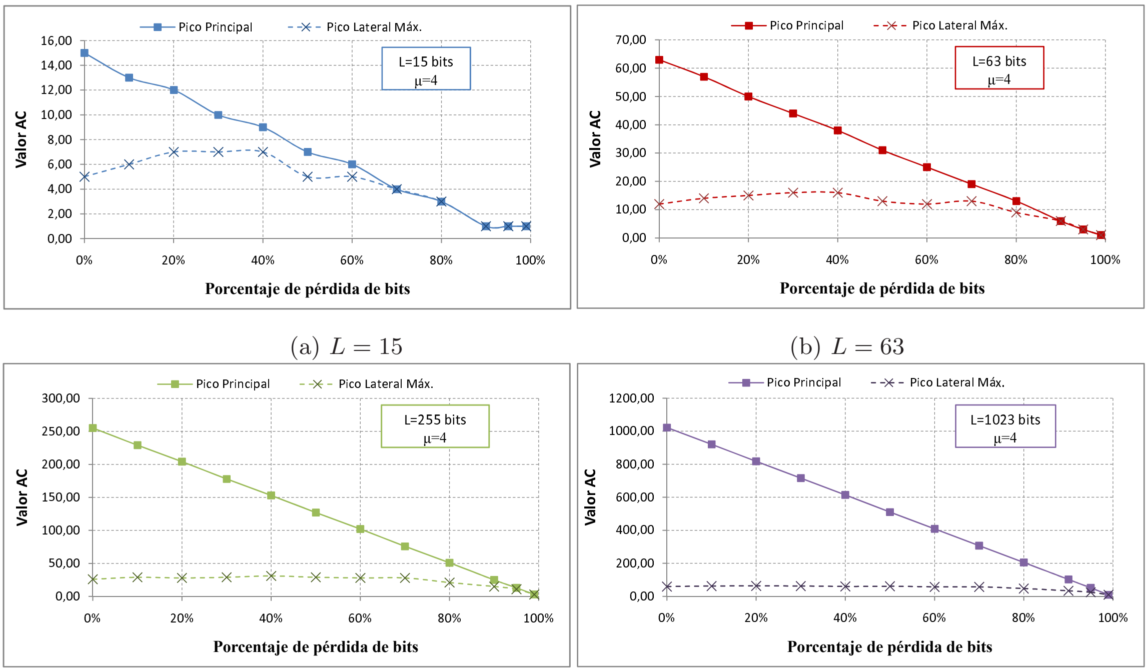


Figura 3.2: θ_{AC} para $\mu = 4$ secuencias Kasami simultáneas en función del porcentaje de secuencia perdido.

3.3. Conjuntos de secuencias complementarias

Los CSS poseen propiedades ideales de correlación cuando se suman las correspondientes funciones de correlación de las secuencias complementarias que los componen. A cada emisor se le asigna un CSS incorrelado con los demás, evitando así interferencias entre ellos. Sin embargo, al codificar cada emisor con más de una secuencia, debe utilizarse un esquema de modulación adecuado que permita la emisión eficiente del mayor número de bits en el menor tiempo posible.

En el caso de parejas Golay (2-CSS) las secuencias pueden emitirse empleando modulaciones QPSK [HUG^{+}04], que en un único símbolo de modulación transmiten los bits las secuencias que componen el par. En [ÁUG^{+}04, Á05] se ha llevado a cabo una intensa búsqueda de un esquema de modulación adecuado para conjuntos de cuatro secuencias complementarias (4-CSS). Así, han considerado esquemas de modulación tales como QPSK doble, 8PSK e incluso un esquema basado en el uso de cuatro símbolos incorrelados (modulación 4D). Se concluye en estos trabajos que, en un sistema de detección donde las señales se demodulan de modo asíncrono mediante correlación con el símbolo de modulación,



(a) $L = 15$ (b) $L = 63$

(c) $L = 255$ (d) $L = 1023$

Figura 3.3: Valores del pico principal de la ACF y del mayor lóbulo lateral en función del porcentaje de secuencia perdido.

el incremento de ganancia de proceso obtenido aumentando el número de bits transmitidos en un ciclo de portadora se ve reducido en la misma cantidad al disminuir la energía transmitida por bit. Esto quiere decir que los resultados obtenidos emitiendo secuencias de $4L$ bits con una modulación 4D son similares a los conseguidos emitiendo en el mismo tiempo secuencias de L bits con una clásica modulación BPSK. Estas conclusiones pueden extrapolarse al caso de conjuntos con mayor número de secuencias, en donde el uso de una modulación BPSK, además de mantener el espectro más estrecho en torno a la frecuencia de la portadora, no supone un incremento notable de la complejidad del esquema de detección eficiente [DMUH$^+$07b] asociado a los CSS emitidos.

Según lo expuesto en el párrafo anterior, el esquema de transmisión consiste en agrupar las secuencias complementarias del conjunto en una única secuencia, conocida como *macro-secuencia*, cuyos bits son emitidos mediante una modulación BPSK. A partir de un conjunto de M secuencias complementarias de longitud L es posible obtener una macro-secuencia $Ms(M, L)$ de longitud $L_{Ms} = M \cdot L$ mediante concatenación de las mismas (Msc), o entrelazando los bits que las componen (Mse).

En (3.13) se representan de forma matricial los bits $\{s_{i,j}[l]; 0 \leq l \leq L-1\}$ que componen las M secuencias $\{s_{i,j}; 0 \leq j \leq M-1\}$ de un conjunto S_i, en donde $0 \leq i \leq L-1$. A partir de esta matriz puede obtenerse una macro-secuencia mediante concatenación (Msc_i) emitiendo una fila tras otra según (3.14); o entrelazando los bits de S_i (Mse_i), lo que equivale a emitir las columnas de la matriz ordenadas una después de otra, como se indica en (3.15).

$$S_i = \begin{pmatrix} s_{i,0} \\ s_{i,1} \\ \vdots \\ s_{i,M-1} \end{pmatrix} = \begin{pmatrix} s_{i,0}[0] & s_{i,0}[1] & \cdots & s_{i,0}[L-1] \\ s_{i,1}[0] & s_{i,1}[1] & \cdots & s_{i,1}[L-1] \\ \vdots & \vdots & \ddots & \vdots \\ s_{i,M-1}[0] & s_{i,M-1}[1] & \cdots & s_{i,M-1}[L-1] \end{pmatrix} \tag{3.13}$$

$$\begin{aligned} Msc_i &= [s_{i,0} \mid s_{i,1} \mid \cdots \mid s_{i,M-2} \mid s_{i,M-1}] \\ &= [s_{i,0}[0]\ s_{i,0}[1]\ \cdots\ s_{i,0}[L-1]\ s_{i,1}[0]\ \cdots\ s_{i,1}[L-1]\ \cdots\ s_{i,M-1}[L-1]] \end{aligned} \tag{3.14}$$

$$\begin{aligned} Mse_i &= [s_{i,0} \otimes s_{i,1} \otimes \cdots \otimes s_{i,M-2} \otimes s_{i,M-1}] \\ &= [s_{i,0}[0]\ s_{i,1}[0]\ \cdots\ s_{i,M-1}[0]\ s_{i,0}[1]\ \cdots\ s_{i,M-1}[1]\ \cdots\ s_{i,M-1}[L-1]] \end{aligned} \tag{3.15}$$

El uso de estas técnicas de emisión mediante macro-secuencias supone la aparición de interferencias en la suma de las funciones de correlación como consecuencia del proceso de detección asíncrono. Al desconocer el instante de llegada de la macro-secuencia emitida no es posible separar convenientemente las secuencias que la componen antes de realizar las correlaciones, apareciendo un fondo de ruido *auto-inducido* aún en ausencia de fuentes de ruido externa [DMUH+06, DM06]:

- *ACF de macro-secuencias de CSS*

  La ACF C_{Msc_i,Msc_i} asociada a una macro-secuencia Msc_i cualquiera generada mediante concatenación puede expresarse en función de las correlaciones del CSS S_i del cual deriva según:

$$\begin{aligned} &C_{Msc_i,Msc_i}[\tau] = \sum_{l=0}^{L_{Msc_i}-1-\tau} Msc_i[l]Msc_i[l+\tau] = \\ &= \sum_{l=0}^{L-1-\tau} (s_{i,0}[l]s_{i,0}[l+\tau] + s_{i,0}[l]s_{i,1}[l-L+\tau] + \cdots + s_{i,0}[l]s_{i,M-1}[l-(M-1)L+\tau] + \\ &+ s_{i,1}[l-L]s_{i,0}[l+\tau] + s_{i,1}[l-L]s_{i,1}[l-L+\tau] + \cdots + s_{i,1}[l-L]s_{i,M-1}[l-(M-1)L+\tau] + \\ &+ \cdots + s_{i,M-1}[l-(M-1)L]s_{i,0}[l+\tau] + s_{i,M-1}[l-(M-1)L]s_{i,1}[l-L+\tau] + \\ &+ \cdots + s_{i,M-1}[l-(M-1)L]s_{i,M-1}[l-(M-1)L+\tau]) \end{aligned} \tag{3.16}$$

$$\begin{aligned} &C_{Msc_i,Msc_i}[\tau] = C_{s_{i,0},s_{i,0}}[\tau] + C_{s_{i,0},si,1}[\tau-L] + \cdots + C_{s_{i,0},s_{i,M-1}}[\tau-(M-1)L] + \\ &+ C_{s_{i,1},s_{i,0}}[\tau+L] + C_{s_{i,1},s_{i,1}}[\tau] + \cdots + C_{s_{i,1},s_{i,M-1}}[\tau-(M-2)L] + \\ &+ \cdots + C_{s_{i,M-1}s_{i,0}}[\tau+(M-1)L] + C_{s_{i,M-1},s_{i,1}}[\tau+(M-2)L] + \cdots + C_{s_{i,M-1},s_{i,M-1}}[\tau] = \\ &= \sum_{j=0}^{M-1} C_{s_{i,j},s_{i,j}}[\tau] + \sum_{j=0}^{M-1} \sum_{\substack{h=0 \\ h \neq j}}^{M-1} C_{s_{i,j},s_{i,h}}[\tau-(h-j)L] \end{aligned} \tag{3.17}$$

  Analizando (3.17) puede observarse que en la ACF de una macro-secuencia Msc_i generada mediante concatenación de las secuencias de un CSS S_i, aparece un primer término con la SACF de todas las secuencias del conjunto S_i de valor $ML \cdot \delta[\tau]$; y un segundo término de interferencia causado por la suma, con diferentes desplazamientos, de las CCF de las secuencias del conjunto.

Del mismo modo, la ACF C_{Mse_i,Mse_i} de una macro-secuencia Mse_i generada mediante entrelazado de los bits que componen las secuencias de un CSS S_i puede expresarse como:

$$\begin{aligned}
&C_{Mse_i,Mse_i}[\tau] = \sum_{l=0}^{L_{Mse_i}-1-\tau} Mse_i[l]Mse_i[l+\tau] = \\
&= \sum_{l=0}^{L} (sv_{i,0}[l]sv_{i,0}[l+\tau] + sv_{i,0}[l]sv_{i,1}[l-1+\tau] + \cdots + sv_{i,0}[l]sv_{i,M-1}[l-(M-1)+\tau]+ \\
&+sv_{i,1}[l-1]sv_{i,0}[l+\tau] + sv_{i,1}[l-1]sv_{i,1}[l-1+\tau] + \cdots + sv_{i,1}[l-1]sv_{i,M-1}[l-(M-1)+\tau]+ \\
&+\cdots + sv_{i,M-1}[l-(M-1)]sv_{i,0}[l+\tau] + sv_{i,M-1}[l+(M-1)]sv_{i,1}[l-1+\tau]+ \\
&+\cdots + sv_{i,M-1}[l-(M-1)]sv_{i,M-1}[l-(M-1)+\tau])
\end{aligned} \tag{3.18}$$

$$\begin{aligned}
&C_{Mse_i,Mse_i}[\tau] = C_{sv_{i,0},sv_{i,0}}[\tau] + C_{sv_{i,0},sv_{i,1}}[\tau-1] + \cdots + C_{sv_{i,0},sv_{i,M-1}}[\tau-(M-1)]+ \\
&+C_{sv_{i,1},sv_{i,0}}[\tau+1] + C_{sv_{i,1},sv_{i,1}}[\tau] + \cdots + C_{sv_{i,1},sv_{i,M-1}}[\tau-(M-2)] + \cdots + \\
&+C_{sv_{i,M-1},sv_{i,0}}[\tau+(M-1)] + C_{sv_{i,M-1},sv_{i,1}}[\tau+(M-2)] + \cdots + C_{sv_{i,M-1},sv_{i,M-1}}[\tau] = \\
&= \sum_{j=0}^{M-1} C_{sv_{i,j},sv_{i,j}}[\tau] + \sum_{j=0}^{M-1} \sum_{\substack{h=0 \\ h\neq j}}^{M-1} C_{sv_{i,j},sv_{i,h}}[\tau-(h-j)]
\end{aligned} \tag{3.19}$$

En donde las secuencias $sv_{i,j}$ representan el producto de Kronecker ($\odot$) de la secuencia $s_{i,j}$ con un vector $1 \times M$ en el que todos sus elementos son cero, excepto el primero que vale 1, véase (3.20). A partir de estas secuencias $sv_{i,j}$ puede construirse fácilmente una macro-secuencia Mse_i como (3.21).

$$\begin{aligned}
sv_{i,j}[\tau] &= s_{i,j} \odot [1\ 0\ 0\ \cdots\ 0]_{1\times M} = \\
&= [s_{i,j}[0]\ (0\cdots 0)_{1\times M-1}\ s_{i,j}[1]\ (0\cdots 0)_{1\times M-1}\ \cdots\ s_{i,j}[L-1]\ (0\cdots 0)_{1\times M-1}]
\end{aligned} \tag{3.20}$$

$$\begin{aligned}
Mse_i[\tau] &= sv_{i,0}[\tau] + sv_{i,1}[\tau-1] + \cdots + sv_{i,M-1}[\tau-(M-1)] = \\
&= [s_{i,0}[0]\ s_{i,1}[0]\ \cdots\ s_{i,M-1}[0]\ s_{i,0}[1]\ \cdots\ s_{i,M-1}[1]\ \cdots\ s_{i,M-1}[L-1]]
\end{aligned} \tag{3.21}$$

En (3.19) se indica que en la ACF de una Mse_i, obtenida a partir de entrelazar los bits de un CSS S_i, aparece un primer término de valor $ML \cdot \delta[\tau]$ debido a la SACF de las secuencias $\{sv_{i,j}; 0 \leq j \leq M-1\}$, que no son más que las secuencias $s_{i,j}$ interpoladas con ceros según (3.20); y un segundo término de interferencia causado por las CCF de las correlaciones de las secuencias $sv_{i,j}$ en distintos desplazamientos.

En la figura 3.4 se muestra una comparativa entre los resultados de correlación obtenidos: (a) emitiendo simultáneamente las 8 secuencias de un conjunto S_0 de longitud $L = 64$ generado con semilla $Wd_2^{(3} = 0$ (véase la sección del B.4 apéndice B); (b) formando una macro-secuencia Msc_0 de $L_{Msc} = 512$ bits, a partir de la concatenación de las secuencias del conjunto; y (c) formando una macro-secuencia

Mse_0 de $L_{Mse} = 512$ bits entrelazando los bits de dichas secuencias. Puede observarse como en (b) y (c) los lóbulos laterales no son nulos, aunque el valor del pico principal es el mismo que el obtenido en (a).

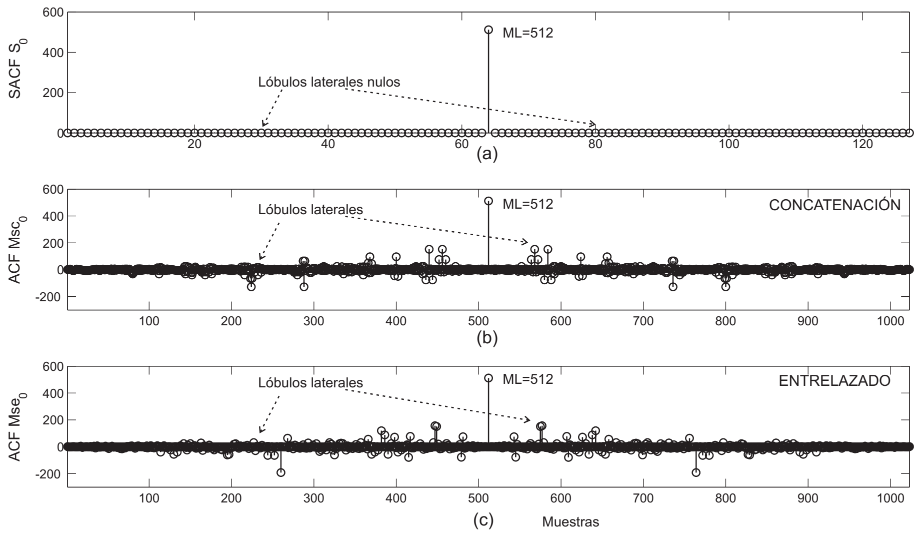


Figura 3.4: Comparativa de las ACF asociadas a un conjunto S_0 de $M = 8$ secuencias de longitud $L = 64$. a) SACF de las 8 secuencias de S_0. b) ACF de una macro-secuencia Msc_0 generada por concatenación de las secuencias de S_0. c) ACF de una macro-secuencia Mse_0 generada por entrelazado de las secuencias de S_0.

- *CCF de macro-secuencias de CSS*

También en la CCF aparecen interferencias consecuencia del mecanismo de ordenación utilizado. Considerando la CCF entre dos macro-secuencias Msc_i y $Msc_{i'}$ generadas a partir de la concatenación de las secuencias de dos CSS S_i y $S_{i'}$ distintos de longitud L, en donde $\{0 \leq i, i' \leq L-1;\ i \neq i'\}$, se tiene:

$$\begin{aligned}
&C_{Msc_i,Msc_{i'}}[\tau] = C_{s_{i,0},s_{i',0}}[\tau] + C_{s_{i,0},s_{i',1}}[\tau - L] + \cdots + C_{s_{i,0},s_{i',M-1}}[\tau - (M-1)L] + \\
&+ C_{s_{i,1}s_{i',0}}[\tau + L] + C_{s_{i,1},s_{i',1}}[\tau] + \cdots + C_{s_{i,1},s_{i',M-1}}[\tau - (M-2)L] + \cdots + \\
&+ C_{s_{i,M-1},s_{i',0}}[\tau + (M-1)L] + C_{s_{i,M-1},s_{i',1}}[\tau + (M-2)L] + \cdots + C_{s_{i,M-1},s_{i',M-1}}[\tau] = \\
&= \sum_{j=0}^{M-1} C_{s_{i,j},s_{i',j}}[\tau] + \sum_{j=0}^{M-1} \sum_{\substack{h=0 \\ h \neq j}}^{M-1} C_{s_{i,j},s_{i',h}}[\tau - (h-j)L]
\end{aligned} \tag{3.22}$$

Análogamente, si las secuencias de los conjuntos S_i y S_i' se emiten mediante entrelazado, la CCF de las macro-secuencias Mse_i y Mse_i' obtenidas queda como:

$$
\begin{aligned}
&C_{Mse_i,Mse_{i'}}[\tau] = C_{sv_{i,0},sv_{i',0}}[\tau] + C_{sv_{i,0},sv_{i',1}}[\tau-1] + \cdots + C_{sv_{i,0},sv_{i',M-1}}[\tau-(M-1)]+\\
&+C_{sv_{i,1},sv_{i',0}}[\tau+1] + C_{sv_{i,1},sv_{i',1}}[\tau] + \cdots + C_{sv_{i,1},sv_{i',M-1}}[\tau-(M-2)] + \cdots +\\
&+C_{sv_{i,M-1},sv_{i',0}}[\tau+(M-1)] + C_{sv_{i,M-1},sv_{i',1}}[\tau+(M-2)] + \cdots + C_{sv_{i,M-1},sv_{i',M-1}}[\tau] =\\
&= \sum_{j=0}^{M-1} C_{sv_{i,j},sv_{i',j}}[\tau] + \sum_{j=0}^{M-1} \sum_{\substack{h=0\\h\neq j}}^{M-1} C_{sv_{i,j},sv_{i',h}}[\tau-(h-j)]
\end{aligned}
\tag{3.23}
$$

De acuerdo a (3.23) y (3.22) la CCF entre dos macro-secuencias distintas depende por un lado de la suma de las CCF entre la j-ésima secuencia del conjunto i y la j-ésima secuencia del conjunto i', cuyo valor es nulo. Por otro lado, es función de la suma de las CCF entre el resto de pares de secuencias que pueden llegar a formarse con las secuencias de cada conjunto (secuencia j-ésima del conjunto S_i con la secuencia h-ésima del conjunto S_i', donde $\{0 \leq j, h \leq M-1; j \neq h\}$), presentando además cada uno de estos pares un desfase distinto. La SCCF de este último término no es nula y por tanto, introduce interferencias. En la figura 3.5 puede observarse este efecto, para el caso de emitir: (a) simultáneamente dos conjuntos S_0 y S_8 con $M = 8$ secuencias de longitud $L = 64$ incorrelados entre ellos; (b) dos macro-secuencias Msc_0 y Msc_8 de longitud $L_{Msc} = 512$ obtenidas concatenando las $M = 8$ secuencias de S_0 y S_8 respectivamente; y (c) dos macro-secuencias Mse_0 y Mse_8 de longitud $L_{Mse} = 512$ obtenidas mediante entrelazado de dichas secuencias. Nótese que el pico de SACF de los conjuntos emitidos y el de ACF de las macro-secuencias tendrían valor 512, como referencia de comparación.

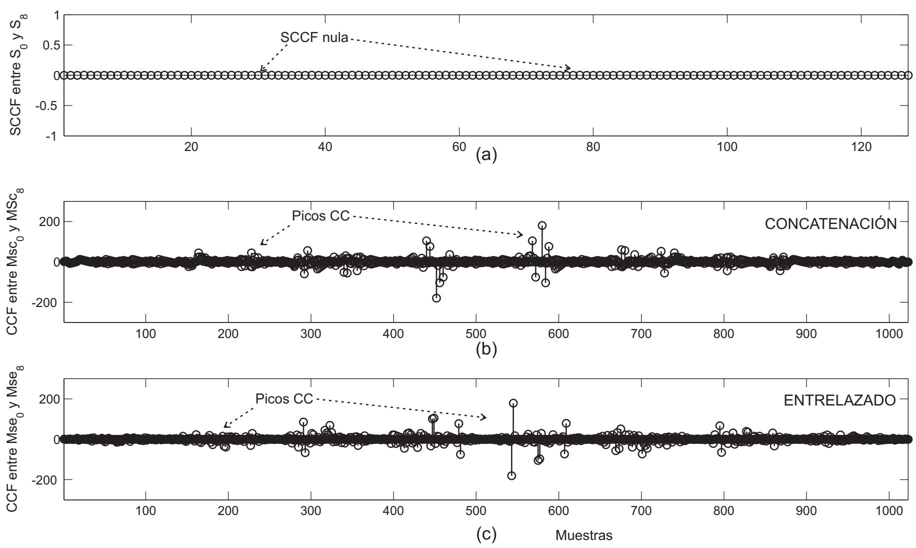


Figura 3.5: Comparativa de las CCF asociadas a dos conjuntos S_0 y S_8 de $M = 8$ secuencias de longitud $L = 64$. a) SCCF entre los conjuntos S_0 y S_8. b) CCF entre dos macro-secuencias Msc_0 y Msc_8 generadas por concatenación. c) CCF entre dos macro-secuencias Mse_0 y Mse_8 generadas por entrelazado.

3.3.1. Selección de macro-secuencias con buenas propiedades de correlación aperiódica

Antes de continuar con el análisis de las macro-secuencias, y para mayor claridad, conviene recordar la nomenclatura utilizada: $Msx_i(M, L)$ indica una macro-secuencia de longitud $L_{Ms} = M \cdot L$ generada mediante concatenación (x=c) o entrelazado (x=e) de un CSS $\{S_i; 0 \leq i \leq L-1\}$; formado por M secuencias $\{s_{i,j}[l]; 0 \leq j \leq M-1; 0 \leq l \leq L-1\}$ de longitud L.

Tanto en el caso de usar concatenación como en el de emplear entrelazado, familias de macro-secuencias de la misma longitud $L_{Ms} = M \cdot L$ pueden obtenerse a partir de diferentes combinaciones de M y L, presentando algunas de ellas interferencias más acusadas que otras (tómese como ejemplo la figura 3.6, donde pueden compararse las ACF de macro-secuencias $Mse_3(2, 128)$ y $Mse_3(16, 16)$ ambas de longitud $L_{Ms} = 256$). Por otro lado, manteniendo los valores de M y L, pero cambiando la semilla de generación del conjunto CSS, se consiguen L macro-secuencias distintas, de modo que para un grupo de usuarios $\mu \leq L$ pueden seleccionarse aquellas que mejores propiedades de correlación presentan. Finalmente, resulta interesante evaluar qué método de ordenación de bits es el más adecuado: concatenación o entrelazado.

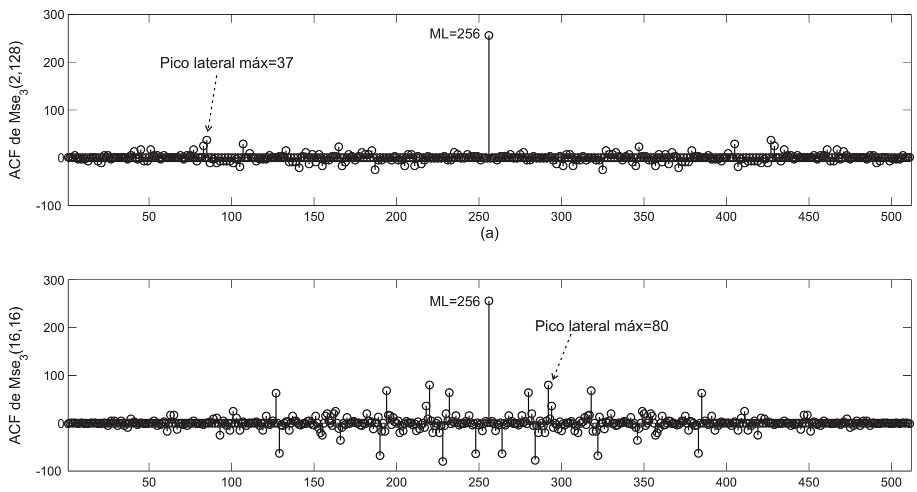


Figura 3.6: a) ACF de una $Mse_3(2, 128)$ de longitud $L_{Ms} = 256$ bits. b) ACF de una $Mse_3(16, 16)$ de longitud $L_{Ms} = 256$ bits.

En [DMUH+06] se realiza una comparativa de macro-secuencias de distinta longitud, generadas mediante concatenación y entrelazado, para el caso particular de cuatro usuarios simultáneos. Se amplía aquí la búsqueda para un rango mayor de longitudes de macro-secuencias y usuarios simultáneos. Para cada caso se proporcionan además las semillas (en formato decimal) de los CSS que dan lugar a las macro-secuencias de menor cota de correlación, obteniendo una lista de gran utilidad para cualquier usuario que opte por el empleo de este esquema de codificación.

En cuanto al número de correlaciones a realizar, en los dos métodos, concatenación y entrelazado, se han tenido en cuenta las posibles variaciones de M y L que dan lugar a macro-secuencias de longitud $L_{MS} = M \cdot L$; y para cada variación se han considerado las L macro-secuencias distintas obtenidas tras variar la semilla del conjunto CSS original. En consecuencia, para cada método y agrupación $M \cdot L$, deben realizarse $\frac{L^2-L}{2}$ correlaciones entre pares de macro-secuencias, eligiendo luego las $\frac{\mu^2-\mu}{2}$ combinaciones de parejas más adecuadas. Además, cabe mencionar que, aunque en emisión aperiódica la cota de correlación obtenida varía en función de la fase de la señal emitida, no se ha considerado aquí esta variación, puesto que los correladores eficientes asociados a los CSS (de uso válido también con macro-secuencias) están acoplados a las secuencias emitidas sin desfasar.

Las cotas teóricas con las que podría resultar interesante comparar las cotas prácticas obtenidas con estas secuencias son la de Levenshtein ($\theta_{Leven.}$) y la de Welch para correlación aperiódica multicanal ($\theta_{Welch-CSS}$). La primera sirve de referencia para su comparación con distintos tipos de códigos binarios $\in \{-1, 1\}$ formados por una única secuencia; y, en este sentido, las macro-secuencias $Ms(M, L)$ pueden considerarse como una nueva familia de L códigos binarios de longitud L_{Ms}, con una sola secuencia por código. Por otro lado, la cota de Welch tiene en cuenta los CSS de los que se derivan las macro-secuencias, y considera las interferencias introducidas en un CSS cuando el número de emisores μ es mayor que el número de secuencias M del CSS original. Aunque esta cota puede resultar de interés para conocer la influencia del uso de un número de CSS mayor que M (y por tanto no incorrelados) en las funciones de correlación de macro-secuencias, sus valores son demasiado optimistas; es decir, proporciona valores muy bajos comparados con los prácticos[1], por lo que no se ha incluido en las tablas de resultados. En cualquier caso, en la búsqueda de macro-secuencias con cotas de correlación mínimas se han evaluado todas las posibilidades y no sólo aquellas generadas a partir de CSS incorrelados. Las tablas 3.3 a 3.10 recogen los resultados obtenidos para cada longitud de macro-secuencia.

$\mathbf{L_{Ms} = 16}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,8)	Concat.	0.3125	0.1875	0.3125	0, 3	–
(2,8)	Entrel.	0.3125	0.3125	0.3125	1, 2	–
(4,4)	Concat.	0.3125	0.3125	0.25	0, 3	–
(4,4)	Entrel.	0.4375	0.3125	0.4375	0, 3	–

[1]Para CSS generados con el método propuesto en [DMUH+07b], se ha calculado la cota de correlación obtenida formando grupos $M < \mu \leq L-1$, obteniendo valores prácticos mayores que los proporcionados por la cota de Welch multi-canal [Wel74].

Continuación de la Tabla 3.3

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 3$						
(2,8)	Concat.	0.5	0.1875	0.5	0, 1, 2	–
(2,8)	Entrel.	0.5	0.1875	0.5	0, 2, 4	–
(4,4)	Concat.	0.5	0.3125	0.5	0, 2, 3	–
(4,4)	Entrel.	0.75	0.3125	0.75	0, 1, 2	–
$\mu = 4$						
(2,8)	Concat.	0.5	0.1875	0.5	0, 1, 2, 3	–
(2,8)	Entrel.	0.5	0.3125	0.5	1, 3, 5, 7	–
(4,4)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3	0.1667
(4,4)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 3	0.1667
$\mu = 5$						
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4	–
(2,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 4, 5	–
$\mu = 6$						
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4, 5	–
(2,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 4, 5, 6	–
$\mu = 7$						
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4, 5, 6	–
(2,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 4, 5, 6, 7	–
$\mu = 8$						
(2,8)	Concat.	0.75	0.1875	0.75	$0, 1, 2, \cdots, 7$	0.1923
(2,8)	Entrel.	0.75	0.3125	0.75	$0, 1, 2, \cdots, 7$	0.1923

Tabla 3.3: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 16$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 32}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,16)	Concat.	0.2188	0.1875	0.2188	0, 8	–
(2,16)	Entrel.	0.2813	0.2188	0.2813	1, 3	–
$\mu = 3$						
(2,16)	Concat.	0.3750	0.1875	0.3750	0, 2, 15	–
(2,16)	Entrel.	0.3750	0.2813	0.3750	5, 6, 13	–

Continuación de la Tabla 3.4

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 4$						
(2,16)	Concat.	0.3750	0.1875	0.3750	0, 2, 13, 15	–
(2,16)	Entrel.	0.3750	0.2813	0.3750	5, 6, 13, 14	–
$\mu = 5$						
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 1, 2, 3, 6	–
(2,16)	Entrel.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7	–
$\mu = 6$						
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 1, 2, 3, 6, 7	–
(2,16)	Entrel.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7, 8	–
$\mu = 7$						
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 2, 3, 6, 7, 11, 15	–
(2,16)	Entrel.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7, 8, 12	–
$\mu = 8$						
(2,16)	Concat.	0.6250	0.1875	0.6250	2, 3, 6, 7, 10, 11, 14, 15	–
(2,16)	Entrel.	0.6250	0.2188	0.6250	0, 3, 4, 7, 8, 11, 12, 15	–
$\mu = 10$						
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 6, 7, 8, 9, 10, 11	–
(2,16)	Entrel.	0.75	0.2188	0.75	0, 1, 3, 4, 7 8, 11, 12, 13, 15	–
$\mu = 12$						
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 6, 7, 8, 9, 10, 11 14, 15	–
(2,16)	Entrel.	0.75	0.2188	0.75	0, 1, 3, 4, 5 7, 8, 9, 11, 12, 13, 15	–
$\mu = 16$						
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, $\cdots$, 15	0.1502
(2,16)	Entrel.	0.75	0.2188	0.75	0, 1, 2, $\cdots$, 15	0.1502

Tabla 3.4: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 32$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 64}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,32)	Concat.	0.2187	0.1094	0.2187	0, 16	–
(2,32)	Entrel.	0.25	0.1719	0.25	2, 14	–
(4,16)	Concat.	0.3125	0.3125	0.1094	0, 12	–
(4,16)	Entrel.	0.375	0.375	0.1719	0, 12	–
(8,8)	Concat.	0.3125	0.3125	0.125	0, 7	–
(8,8)	Entrel.	0.3125	0.3125	0.2344	0, 7	–
$\mu = 3$						
(2,32)	Concat.	0.3125	0.1563	0.3125	0, 3, 12	–
(2,32)	Entrel.	0.3594	0.1719	0.3594	1, 2, 13	–
(4,16)	Concat.	0.3125	0.3125	0.25	0, 3, 12	–
(4,16)	Entrel.	0.375	0.375	0.3125	0, 5, 11	–
(8,8)	Concat.	0.3125	0.3125	0.25	0, 3, 5	–
(8,8)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5	–
$\mu = 4$						
(2,32)	Concat.	0.3125	0.1563	0.3125	0, 3, 12, 15	–
(2,32)	Entrel.	0.3594	0.1719	0.3594	1, 2, 13, 14	–
(4,16)	Concat.	0.3125	0.3125	0.25	0, 3, 12, 15	–
(4,16)	Entrel.	0.375	0.375	0.3125	0, 5, 10, 15	–
(8,8)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6	–
(8,8)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6	–
$\mu = 5$						
(2,32)	Concat.	0.4375	0.1094	0.4375	0, 2, 9, 11, 16	–
(2,32)	Entrel.	0.4375	0.1719	0.4375	1, 2, 13, 14, 19	–
(4,16)	Concat.	0.3438	0.3125	0.3438	0, 3, 8, 11, 12	–
(4,16)	Entrel.	0.375	0.375	0.375	0, 3, 8, 11, 12	–
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4	–
(8,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 3, 4	–
$\mu = 6$						
(2,32)	Concat.	0.4375	0.1094	0.4375	0, 2, 9, 11, 16, 18	–
(2,32)	Entrel.	0.4375	0.1719	0.4375	1, 3, 5, 10, 12, 14	–
(4,16)	Concat.	0.3438	0.3125	0.3438	0, 3, 8, 11, 12, 15	–
(4,16)	Entrel.	0.375	0.375	0.375	0, 7, 8, 11, 12, 15	–
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5	–
(8,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5	–
$\mu = 7$						

Continuación de la Tabla 3.5

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(2,32)	Concat.	0.4375	0.1563	0.4375	0, 2, 4, 6, 9, 11, 13	–
(2,32)	Entrel.	0.4375	0.1719	0.4375	1, 3, 5, 10, 12, 26, 28	–
(4,16)	Concat.	0.4063	0.3125	0.4063	0, 3, 4, 7, 8, 11, 12	–
(4,16)	Entrel.	0.3906	0.375	0.3906	1, 2, 5, 9, 10, 13, 14	–
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6	–
(8,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5, 6	–
$\mu = 8$						
(2,32)	Concat.	0.4375	0.1563	0.4375	0, 2, 4, 6, 9, 11, 13, 15	–
(2,32)	Entrel.	0.4375	0.1719	0.4375	1, 3, 10, 12, 17, 20, 26, 28	–
(4,16)	Concat.	0.4063	0.3125	0.4063	0, 3, 4, 7, 8, 11, 12, 15	–
(4,16)	Entrel.	0.4219	0.375	0.4219	1, 2, 5, 6, 9, 10, 13, 14	–
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, $\cdots$ 6	0.0974
(8,8)	Entrel.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 7	0.0974
$\mu = 10$						
(2,32)	Concat.	0.625	0.1094	0.625	0, 1, 2, 3, 8, 9, 10, 11, 16, 17	–
(2,32)	Entrel.	0.625	0.1719	0.625	1, 2, 3, 4, 5, 6, 9, 10, 11, 12	–
(4,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4 5, 6, 7, 8, 9	–
(4,16)	Entrel.	0.5156	0.375	0.5156	0, 1, 2, 3, 4 5, 6, 7, 8, 9	–
$\mu = 16$						
(2,32)	Concat.	0.625	0.1094	0.625	0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19, 24, 25, 26, 27	–
(2,32)	Entrel.	0.625	0.1719	0.625	1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 19, 20, 27, 28	–
(4,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	0.1062
(4,16)	Entrel.	0.5156	0.375	0.5156	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	0.1062

Continuación de la Tabla 3.5

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 32$						
(2,32)	Concat.	0.75	0.1562	0.75	0, 1, 2, $\cdots$, 31	0.1120
(2,32)	Entrel.	0.75	0.2031	0.75	0, 1, 2, $\cdots$, 31	0.1120

Tabla 3.5: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 64$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 128}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,64)	Concat.	0.1484	0.1406	0.1484	8, 40	–
(2,64)	Entrel.	0.1953	0.1641	0.1953	6, 38	–
$\mu = 3$						
(2,64)	Concat.	0.2578	0.1094	0.2578	0, 2, 37	–
(2,64)	Entrel.	0.2578	0.1484	0.2578	0, 37, 39	–
$\mu = 4$						
(2,64)	Concat.	0.2578	0.1094	0.2578	0, 2, 37, 39	–
(2,64)	Entrel.	0.2734	0.1797	0.2734	0, 2, 37, 39	–
$\mu = 5$						
(2,64)	Concat.	0.3125	0.1094	0.3125	0, 2, 13, 15, 21	–
(2,64)	Entrel.	0.3203	0.1797	0.3203	7, 18, 25, 39, 50	–
$\mu = 6$						
(2,64)	Concat.	0.3125	0.1094	0.3125	0, 2, 13, 15, 21, 23	–
(2,64)	Entrel.	0.3203	0.1797	0.3203	7, 18, 25, 39, 50, 57	–
$\mu = 7$						
(2,64)	Concat.	0.3125	0.1406	0.3125	0, 2, 13, 15, 21, 23, 24	–
(2,64)	Entrel.	0.3359	0.1797	0.3359	0, 2, 13, 15, 21, 23, 24	–
$\mu = 8$						
(2,64)	Concat.	0.3125	0.1406	0.3125	0, 2, 13, 15, 21, 23, 24, 26	–
(2,64)	Entrel.	0.3516	0.1797	0.3516	0, 2, 13, 15, 21, 23, 24, 26	–
$\mu = 10$						
(2,64)	Concat.	0.4063	0.1406	0.4063	0, 13, 19, 21, 23, 24, 26, 30, 52, 58	–

Continuación de la Tabla 3.6

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(2,64)	Entrel.	0.4063	0.1797	0.4063	1, 22, 25, 27, 31, 34, 53, 56, 58, 60	–
$\mu = 64$						
(2,64)	Concat.	0.75	0.1406	0.75	0, 1, 2, $\cdots$, 63	0.082
(2,64)	Entrel.	0.75	0.1953	0.75	0, 1, 2, $\cdots$, 63	0.082

Tabla 3.6: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 128$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 256}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,128)	Concat.	0.1328	0.0859	0.1328	12, 76	–
(2,128)	Entrel.	0.1601	0.1367	0.1602	0, 64	–
(4,64)	Concat.	0.1953	0.1953	0.0938	4, 18	–
(4,64)	Entrel.	0.1914	0.1914	0.0977	4, 11	–
(16,16)	Concat.	0.3125	0.3125	0.0625	0, 15	–
(16,16)	Entrel.	0.3125	0.3125	0.1211	0, 15	–
$\mu = 3$						
(2,128)	Concat.	0.2188	0.0859	0.2188	8, 54, 79	–
(2,128)	Entrel.	0.2305	0.1289	0.2305	4, 74, 125	–
(4,64)	Concat.	0.1953	0.1953	0.1875	0, 11, 20	–
(4,64)	Entrel.	0.2188	0.1914	0.2188	0, 11, 20	–
(16,16)	Concat.	0.2656	0.2656	0.25	8, 11, 13	–
(16,16)	Entrel.	0.3125	0.3125	0.2383	0, 7, 13	–
$\mu = 4$						
(2,128)	Concat.	0.2344	0.0859	0.2344	8, 54, 79, 113	–
(2,128)	Entrel.	0.2344	0.1211	0.2344	13, 62, 77, 126	–
(4,64)	Concat.	0.1953	0.1953	0.1875	0, 11, 20, 31	–
(4,64)	Entrel.	0.2188	0.1992	0.2188	0, 11, 20, 31	–
(16,16)	Concat.	0.2656	0.2656	0.25	8, 11, 13, 14	–
(16,16)	Entrel.	0.3125	0.3125	0.2461	3, 4, 8, 15	–
$\mu = 5$						
(2,128)	Concat.	0.2578	0.1016	0.2578	8, 57, 66, 70, 115	–
(2,128)	Entrel.	0.2813	0.1211	0.2813	1, 17, 42, 52, 115	–
(4,64)	Concat.	0.25	0.1953	0.25	0, 3, 12, 15, 32	–
(4,64)	Entrel.	0.25	0.1992	0.25	1, 13, 33, 36, 40	–

Continuación de la Tabla 3.7

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9	–
(16,16)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9	–
$\mu = 6$						
(2,128)	Concat.	0.2578	0.0977	0.2578	16, 22, 105, 107, 109, 111	–
(2,128)	Entrel.	0.2813	0.1289	0.2813	8, 10, 24, 45, 61, 63	–
(4,64)	Concat.	0.25	0.1953	0.25	0, 3, 12, 15, 32, 35	–
(4,64)	Entrel.	0.2617	0.1992	0.2617	1, 3, 5, 9, 10, 12	–
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10	–
(16,16)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10	–
$\mu = 7$						
(2,128)	Concat.	0.2813	0.0977	0.2813	4, 9, 11, 20, 25, 27, 46	–
(2,128)	Entrel.	0.2813	0.1367	0.2813	8, 10, 24, 45, 47, 61, 63	–
(4,64)	Concat.	0.2578	0.1953	0.2578	0, 3, 5, 6, 9, 10, 12	–
(4,64)	Entrel.	0.2617	0.1992	0.2617	1, 3, 5, 9, 10, 12, 15	–
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12	–
(16,16)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12	–
$\mu = 8$						
(2,128)	Concat.	0.2813	0.1016	0.2813	0, 13, 15, 16, 29, 31, 42, 58	–
(2,128)	Entrel.	0.2930	0.1445	0.2930	9, 11, 25, 27, 44, 46, 60, 62	–
(4,64)	Concat.	0.2578	0.1953	0.2578	0, 3, 5, 6, 9, 10, 12, 15	–
(4,64)	Entrel.	0.2617	0.207	0.2617	0, 3, 5, 6, 9, 10, 12, 15	–
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12, 15	–
(16,16)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12, 15	–
$\mu = 10$						
(2,128)	Concat.	0.3125	0.1016	0.3125	0, 2, 4, 6, 65, 67, 109, 111, 125, 127	–
(2,128)	Entrel.	0.3281	0.1289	0.3281	4, 6, 11, 25, 39, 42, 55, 56, 58, 69	–

Continuación de la Tabla 3.7

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(16,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5 6, 7, 8, 9	–
(16,16)	Entrel.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5 6, 7, 8, 9	–
$\mu = 16$						
(16,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, $\cdots$, 15	0.0531
(16,16)	Entrel.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 15	0.0531
$\mu = 64$						
(4,64)	Concat.	0.75	0.1953	0.75	0, 1, 2, $\cdots$, 63	0.0580
(4,64)	Entrel.	0.75	0.2070	0.75	0, 1, 2, $\cdots$, 63	0.0580
$\mu = 128$						
(2,128)	Concat.	0.75	0.1016	0.75	0, 1, 2, $\cdots$, 127	0.0594
(2,128)	Entrel.	0.75	0.1445	0.75	0, 1, 2, $\cdots$, 127	0.0594

Tabla 3.7: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 256$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 512}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,256)	Concat.	0.0977	0.082	0.0977	24, 152	–
(2,256)	Entrel.	0.1348	0.1152	0.1348	50, 178	–
(8,64)	Concat.	0.3281	0.3281	0.0527	16, 40	–
(8,64)	Entrel.	0.375	0.375	0.0723	16, 40	–
$\mu = 3$						
(2,256)	Concat.	0.1797	0.0742	0.1797	64, 199, 233	–
(2,256)	Entrel.	0.1934	0.0996	0.1934	8, 23, 153	–
(8,64)	Concat.	0.3125	0.3125	0.1172	8, 19, 61	–
(8,64)	Entrel.	0.375	0.375	0.1328	0, 42, 63	–
$\mu = 4$						
(2,256)	Concat.	0.1914	0.082	0.1914	36, 60, 175, 183	–
(2,256)	Entrel.	0.1953	0.1035	0.1953	14, 17, 36, 59	–
(8,64)	Concat.	0.3125	0.3125	0.1328	8, 16, 31, 52	–
(8,64)	Entrel.	0.375	0.375	0.1465	19, 24, 39, 44	–
$\mu = 5$						
(2,256)	Concat.	0.2187	0.0742	0.2188	0, 23, 31, 148, 156	–

Continuación de la Tabla 3.8

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(2,256)	Entrel.	0.2187	0.1074	0.2188	4, 12, 19, 148, 156	–
(8,64)	Concat.	0.2969	0.2969	0.25	1, 18, 21, 24, 27	–
(8,64)	Entrel.	0.375	0.375	0.3535	1, 4, 6, 7, 9	–
$\mu = 6$						
(2,256)	Concat.	0.2187	0.0898	0.2188	0, 8, 23, 31, 148, 156	–
(2,256)	Entrel.	0.2344	0.0996	0.2344	36, 73, 119, 164, 201, 23	–
$\mu = 7$						
(2,256)	Concat.	0.2344	0.082	0.2344	0, 12, 39, 43, 213, 217, 242	–
(2,256)	Entrel.	0.2344	0.123	0.2344	4, 12, 27, 102, 143, 144, 242	–
$\mu = 8$						
(2,256)	Concat.	0.2422	0.082	0.2422	12, 39, 85, 126, 140, 167, 213, 254	–
(2,256)	Entrel.	0.2441	0.1151	0.2441	14, 45, 81, 96, 146, 163, 223, 252	–
$\mu = 64$						
(8,64)	Concat.	0.5	0.3281	0.5	0, 1, 2, $\cdots$, 63	0.0410
(8,64)	Entrel.	0.5020	0.3750	0.5020	0, 1, 2, $\cdots$, 63	0.0410
$\mu = 256$						
(2,256)	Concat.	0.75	0.0977	0.75	0, 1, 2, $\cdots$, 255	0.0426
(2,256)	Entrel.	0.75	0.1269	0.75	0, 1, 2, $\cdots$, 255	0.0426

Tabla 3.8: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 512$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 1024}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,512)	Concat.	0.0811	0.0684	0.0811	240, 496	–
(2,512)	Entrel.	0.1064	0.1064	0.1064	60, 316	–
(32,32)	Concat.	0.3125	0.3125	0.0313	0, 31	–
(32,32)	Entrel.	0.3125	0.3125	0.0615	0, 31	–
$\mu = 3$						

Continuación de la Tabla 3.9

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(2,512)	Concat.	0.1484	0.0645	0.1484	32, 223, 229	–
(2,512)	Entrel.	0.1484	0.0908	0.1484	65, 123, 132	–
(32,32)	Concat.	0.2539	0.2539	0.25	24, 29, 30	–
(32,32)	Entrel.	0.3125	0.3125	0.1943	0, 14, 23	–
$\mu = 4$						
(2,512)	Concat.	0.1523	0.0645	0.1523	172, 223, 428, 479	–
(2,512)	Entrel.	0.1523	0.0850	0.1523	173, 222, 429, 478	–
(32,32)	Concat.	0.2539	0.2539	0.25	24, 27, 29, 30	–
(32,32)	Entrel.	0.3125	0.3125	0.2178	5, 10, 22, 25	–
$\mu = 5$						
(2,512)	Concat.	0.1758	0.0645	0.1758	92, 102, 201, 378, 469	–
(2,512)	Entrel.	0.1758	0.0811	0.1758	28, 64, 151, 314, 407	–
(32,32)	Concat.	0.2656	0.2656	0.25	16, 25, 26, 28, 31	–
(32,32)	Entrel.	0.3125	0.3125	0.249	1, 12, 20, 24, 31	–
$\mu = 6$						
(2,512)	Concat.	0.1885	0.0684	0.1885	0, 84, 169, 253, 319, 363	–
(2,512)	Entrel.	0.1914	0.0908	0.1914	78, 81, 100, 123, 338, 359	–
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26	–
(32,32)	Entrel.	0.3125	0.3125	0.249	0, 7, 11, 12, 20, 25	–
$\mu = 7$						
(2,512)	Concat.	0.1914	0.0684	0.1914	4, 80, 173, 249, 315,	–
(2,512)	Entrel.	0.2031	0.085	0.2031	33, 92, 136, 159, 245, 342, 386	–
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26, 28	–
(32,32)	Entrel.	0.3125	0.3125	0.2568	1, 7, 11, 12, 19, 20, 24	–
$\mu = 8$						
(2,512)	Concat.	0.1992	0.0684	0.1992	8, 24, 198, 217, 267, 283, 453, 474	–
(2,512)	Entrel.	0.2070	0.083	0.207	8, 45, 50, 231, 267, 302, 305, 484	–
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26, 28, 31	–

Continuación de la Tabla 3.9

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(32,32)	Entrel.	0.3125	0.3125	0.2568	1, 7, 11, 12, 19, 20, 24, 31	–
$\mu = 10$						
(32,32)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12, 15, 17, 18	–
(32,32)	Entrel.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12, 15, 17, 18	–
$\mu = 32$						
(32,32)	Concat.	0.5	0.3125	0.5	$0, 1, 2, \cdots, 31$	0.0280
(32,32)	Entrel.	0.75	0.3125	0.75	$0, 1, 2, \cdots, 31$	0.0280
$\mu = 512$						
(2,512)	Concat.	0.75	0.0727	0.75	$0, 1, 2, \cdots, 255$	0.0305
(2,512)	Entrel.	0.75	0.1064	0.75	$0, 1, 2, \cdots, 255$	0.0305

Tabla 3.9: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 1024$, en función del número μ de usuarios simultáneos y método de emisión.

$\mathbf{L_{Ms} = 2048}$

(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
$\mu = 2$						
(2,1024)	Concat.	0.0674	0.0576	0.0674	200, 712	–
(2,1024)	Entrel.	0.0835	0.0649	0.0835	109, 621	–
$\mu = 3$						
(2,1024)	Concat.	0.1230	0.0518	0.123	40, 215, 637	–
(2,1024)	Entrel.	0.123	0.0659	0.123	6, 249, 595	–
$\mu = 4$						
(2,1024)	Concat.	0.1230	0.0557	0.1230	8, 355, 520, 867	–
(2,1024)	Entrel.	0.1284	0.0591	0.1284	163, 316, 675, 828	–
$\mu = 5$						
(2,1024)	Concat.	0.1484	0.0576	0.1484	36, 134, 629, 911, 1019	–
(2,1024)	Entrel.	0.1523	0.0728	0.1523	109, 306, 464, 643, 799	–
$\mu = 6$						
(2,1024)	Concat.	0.1543	0.0557	0.1543	32, 317, 475, 544, 829, 987	–

Continuación de la Tabla 3.10

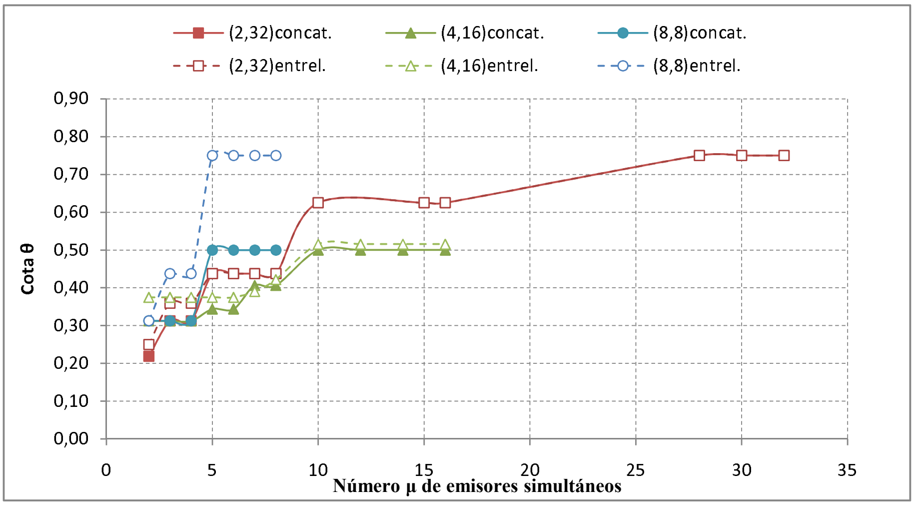
(M,L)	Método	θ	θ_{AC}	θ_{CC}	semillas	$\theta_{Leven.}$
(2,1024)	Entrel.	0.1602	0.0747	0.1602	3, 41, 348, 543, 565, 832	–
$\mu = 1024$						
(2,1024)	Concat.	0.75	0.0635	0.75	0, 1, 2, $\cdots$, 1023	0.0217
(2,1024)	Entrel.	0.75	0.0874	0.75	0, 1, 2, $\cdots$, 1023	0.0217

Tabla 3.10: Cotas aperiódicas mínimas para macro-secuencias $Ms(M, L)$ longitud $L_{MS} = 2048$, en función del número μ de usuarios simultáneos y método de emisión.

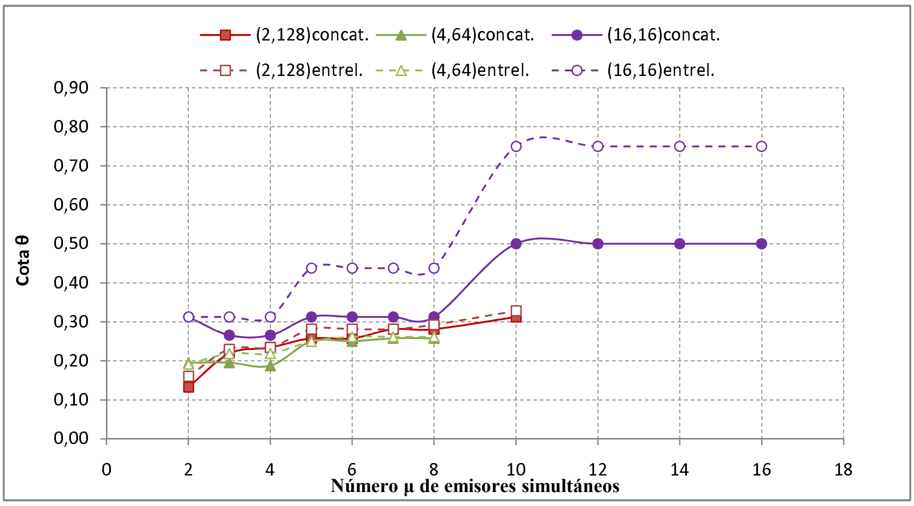
Para mayor claridad se han representado de forma gráfica los valores de algunas de estas tablas. En concreto, la figura 3.7 muestra las cotas mínimas de correlación obtenidas con macro-secuencias de longitud 64 (a) y longitud 256 (b) en función del número de usuarios simultáneos. Aquí puede observarse como, a diferencia con las secuencias Kasami, para una misma longitud de macro-secuencia un aumento del número de usuarios implica cotas más altas. La figura 3.8 resume las cotas de correlación obtenidas para el caso de 5 (a) y 8 (b) usuarios simultáneos, pudiéndose apreciar como se obtienen cotas más bajas con secuencias más largas. De ambas figuras se deduce que la cota de correlación es menor cuando los bits del CSS se ordenan mediante concatenación, ahora bien, esto es más acusado cuando las macro-secuencias se forman a partir de CSS con $M = L$. Sin embargo cuando $L > M$, esto es, cuando el generador de CSS está formado por más de una etapa (véase B.4), esta diferencia es menor. En los casos que $L > M$ y, sobre todo, para longitudes de macro-secuencia L_{Ms} elevadas, es preferible el uso de entrelazado ya que variaciones rápidas del canal de transmisión afectarán por igual a todas las secuencias, reduciendo los efectos negativos sobre la complementaridad de los CSS. Se observa además que, independientemente del esquema de transmisión, las peores cotas se obtienen para macro-secuencias generadas a partir de CSS cuya longitud coincide con el número de secuencias ($M = L$). Por último, resulta evidente que las macro-secuencias se ajustan poco a la cota de Levenshtein, debido principalmente a los valores altos de correlación cruzada obtenidos cuando el número de usuarios simultáneos es el máximo posible.

3.3.2. Influencia de la recepción parcial de la macro-secuencia en la cota de auto-correlación

Conviene también analizar qué método de ordenación de bits permite la detección más clara del eco emitido cuando se usa un único transductor como emisor y receptor, y el reflector se haya muy próximo al transductor. En la figura 3.9 se ha representado la θ_{AC} obtenida al correlar $\mu = 4$ macro-secuencias de distintas longitudes en función del porcentaje de secuencia perdido. Las macro-secuencias no se han elegido de modo arbitrario, sino que

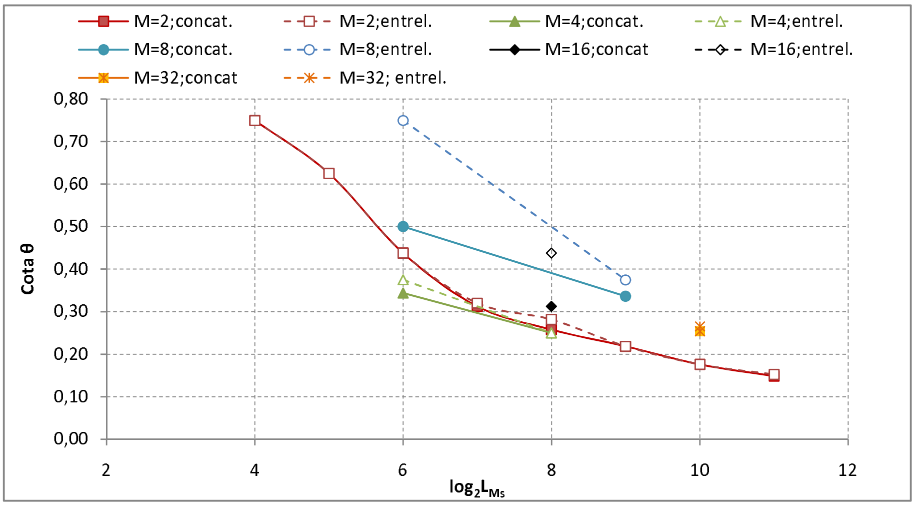


(a) $L_{Ms} = 64$

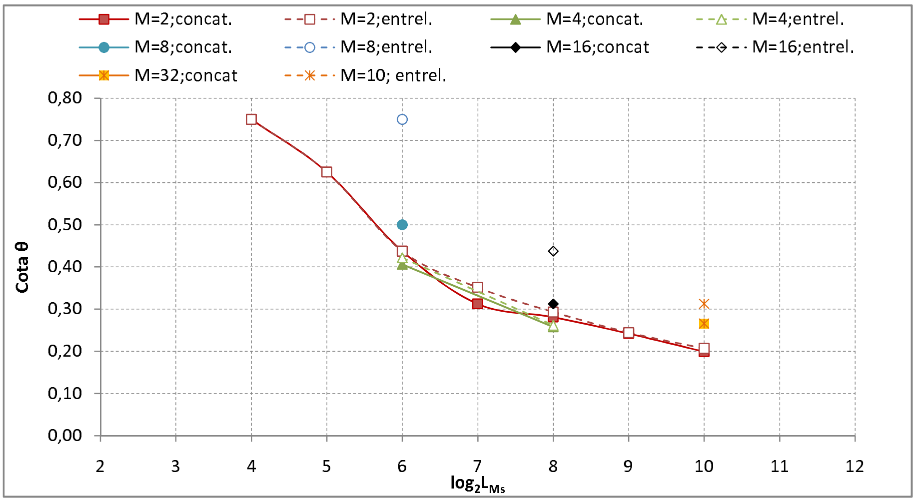


(b) $L_{Ms} = 256$

Figura 3.7: Cotas de correlación mínima obtenidas con macro-secuencias $Ms(M, L)$ de longitud a) $L_{Ms} = 64$ y b) $L_{Ms} = 256$ en función del número μ de usuarios simultáneos.



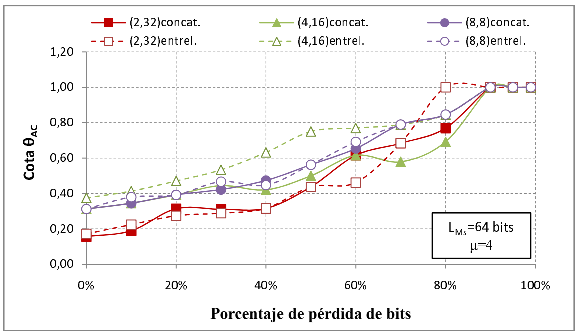
(a) $\mu = 5$



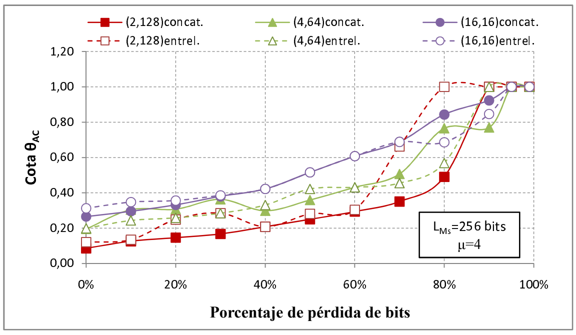
(b) $\mu = 8$

Figura 3.8: Cotas de correlación para el caso de emitir a) $\mu = 5$ y b) $\mu = 8$ macrosecuencias $Ms(M, L)$ simultáneas de distinta longitud L_{Ms}, en función del número M de secuencias del CSS.

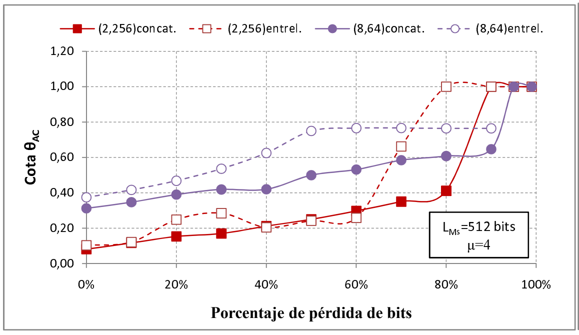
se han considerado las de menor cota de correlación según el estudio realizado en la sección previa. Como puede verse en esta figura los mejores resultados se han obtenido cuando el número de secuencias del conjunto es el menor posible ($M = 2$). Por ejemplo, considerando longitudes de secuencia elevadas ($L_{Ms} \geq 256$), la θ_{AC} obtenida con macro-secuencias generadas mediante concatenación de parejas Golay ($M = 2$) es menor que 0.5 siempre que el porcentaje de secuencia perdido no supere el 80 %; mientras que para mantener $\theta_{AC} = 0.5$ con cuatro macro-secuencias generadas mediante concatenación de conjuntos con el mayor número M de secuencias (con $L_{Ms} = M \cdot L$ constante), el porcentaje perdido no puede exceder el 50 %. Por otro lado, puede observarse que, en general, el método de emisión mediante concatenación de CSS proporciona mejores resultados de cota de auto-correlación ante la pérdida de bits al inicio de la secuencia.



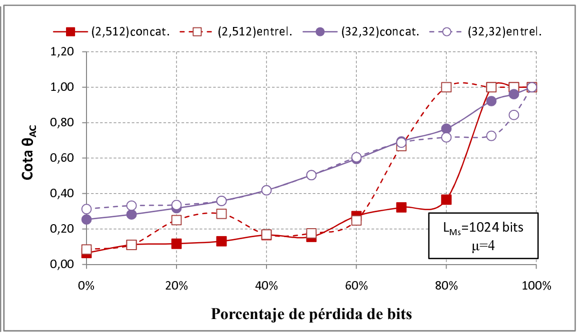
(a) $L_{Ms} = 64$



(b) $L_{Ms} = 256$



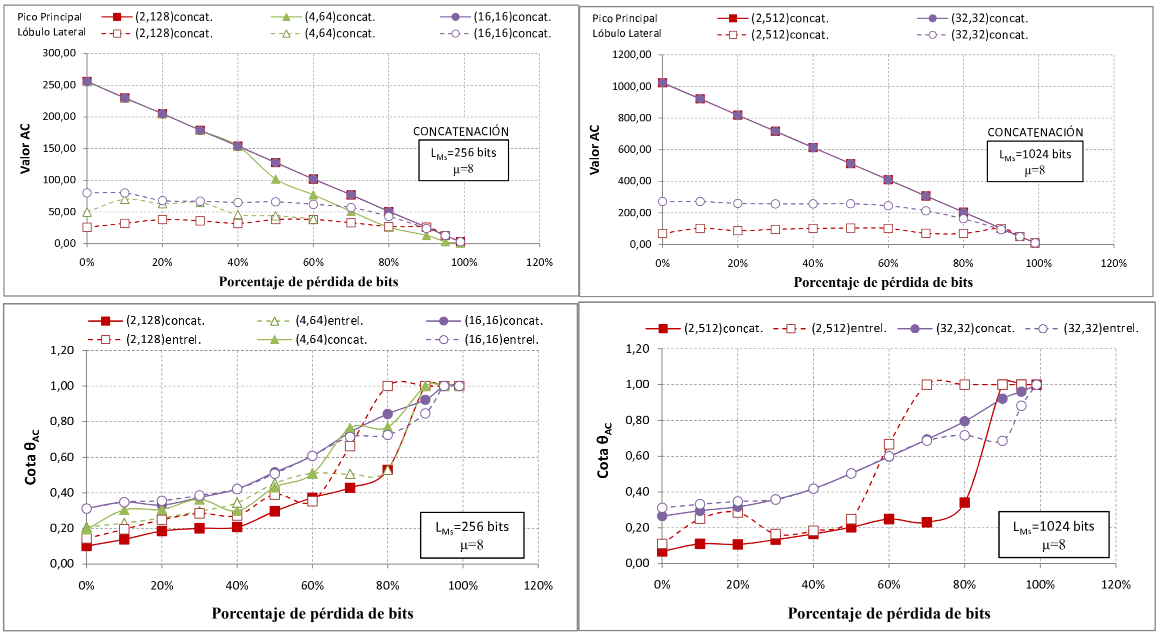
(c) $L_{Ms} = 512$



(d) $L_{Ms} = 1024$

Figura 3.9: θ_{AC} para $\mu = 4$ macro-secuencias simultáneas en función del porcentaje de secuencia perdido.

Resultados parecidos se muestran en la figura 3.10 para macro-secuencias de 256 y 1024 bits cuando el número de emisores simultáneos es $\mu = 8$.



(a) $L_{Ms} = 256$ (b) $L_{Ms} = 1024$

Figura 3.10: Comportamiento de grupos de $\mu = 8$ macro-secuencias simultáneas en función del porcentaje de secuencia perdido.

3.4. Códigos LS

Los códigos LS pertenecen al grupo de códigos ortogonales generalizados, así en su ACF y CCF puede encontrarse una ventana libre de interferencias (IFW) alrededor del origen en donde la cota de correlación $\theta = 0$, en tanto en cuanto la diferencia de llegada de los distintos códigos esté dentro de la IFW. Estas secuencias están pensadas para trabajar en entornos cuasi-síncronos, de modo que es posible eliminar las interferencias ISI y MAI aún cuando existen pequeños retardos entre las recepciones debidos, por ejemplo, a una mala sincronización entre emisiones, a la propagación multicamino o al propio retardo que introduce el medio. Cuando el sistema opera de modo asíncrono deben emplearse técnicas que permitan asegurar que la diferencia de los TDV no excede la IFW, ya que las interferencias fuera de esa zona pueden ser elevadas, según se observa en la figura 3.11. En un sistema de posicionamiento local ultrasónico asíncrono esto puede conseguirse eligiendo adecuadamente la disposición de los emisores y acotando la zona de posicionamiento.

Se pretende en esta sección caracterizar el comportamiento de los códigos LS fuera de la IFW. Teniendo en mente este objetivo se llevará a cabo una selección de las secuencias LS con menor cota de correlación θ y número de valores no nulos fuera de la IFW, de modo que se reduzcan al máximo las interferencias en caso de que algún eco se reciba dentro del área con interferencias (IW, *Interference Window*). Además, al igual que se ha hecho con las secuencias Kasami y las macro-secuencias de CSS, se estudiará en cuánto puede reducirse el tamaño de la zona ciega cuando se usa un único transductor para la emisión y recepción

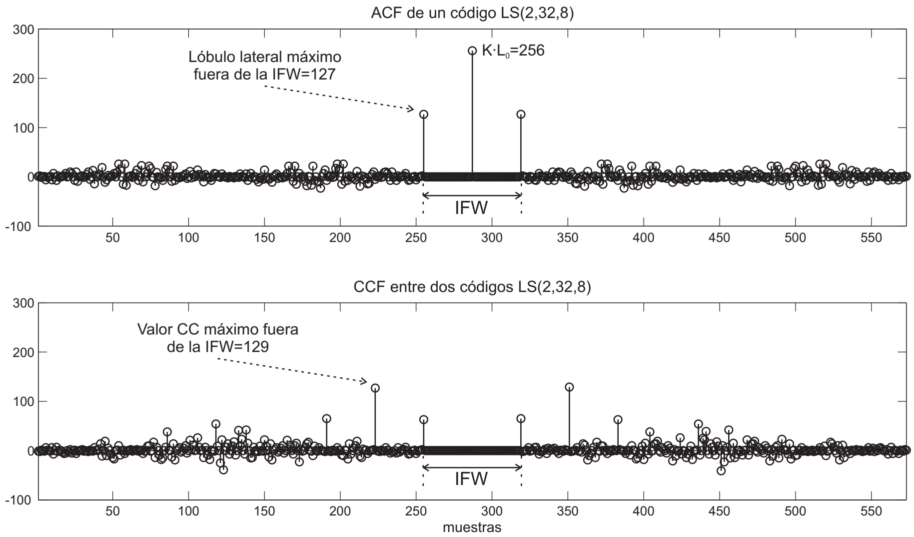


Figura 3.11: ACF y CCF de códigos LS(2,32,8) de longitud $L = 287$ bits, generados a partir de CSS con $M = 2$ secuencias de longitud $L_0 = 32$.

de códigos LS.

En la sección 2.2.6 del capítulo 2 se detallan dos métodos con los que obtener códigos LS, a partir de parejas Golay [SBH01] o CSS [ZYH05][2]. Ambos métodos son similares: cuando se usan pares Golay (2-CSS) la diferencia viene dada principalmente porque en el primero se repiten varias veces las secuencias del par que corresponda, obteniendo una familia con un número K de códigos cualquiera $K = 2^n,\ n \in \mathbb{N}$; y en el segundo, las secuencias de cada par aparecen una única vez en el código LS, estando el número de códigos de la familia restringido a $K = M^2$. Se obtienen por tanto los mismos resultados con los dos métodos cuando $M = 2$ y $K = 4$.

Antes de comenzar el estudio de interferencias en la IW se recuerda la nomenclatura:

- M: nº de secuencias de un conjunto complementario. Cuando se trabaja con pares Golay $M = 2$.
- L_0: longitud de las secuencias del CSS.
- W: tamaño de la semi-ventana libre de interferencias. Normalmente $W = L_0 - 1$.
- K: número de códigos de una familia LS. $K = 2^n,\ n \in \mathbb{N}$ si las secuencias LS se generan según [SBH01]; y $K = M^2$ cuando se generan según [ZYH05].
- L: longitud de las secuencias LS. $L = K \cdot L_0 + W$ si las secuencias LS se generan según [SBH01]; y $L = K \cdot L_0 + (M-1)W$ cuando se generan según [ZYH05].

No se muestra para estas secuencias ninguna cota teórica, ya que las clásicas de Welch, Sarwate o Levenshtein no se ajustan bien a estas secuencias; y las propuestas por Tang-Fan

[2]Considerando $P = 1$ en el algoritmo propuesto en [ZYH05]; véase la sección 2.2.6 para más detalles.

y Peng-Fan proporcionan cotas mínimas de correlación dentro de la IFW cuando se trabaja con códigos cuasi-ortogonales generalizados. Además, sólo se han evaluado secuencias LS sin desfasar, puesto que desfasarlas supondría desfasar también los CSS (considerando los pares Golay como 2-CSS) de los cuales derivan, y los correladores eficientes asociados a estos CSS dejarían de ser de utilidad. Estas mismas consideraciones pueden aplicarse también a códigos T-ZCZ.

3.4.1. Estudio de la IW asociada a códigos LS generados a partir de pares Golay incorrelados

Se han hecho pruebas con todos los pares Golay incorrelados posibles de la misma longitud hasta encontrar aquellos que implican magnitudes menores de interferencias en la IW de secuencias LS. Hay que decir, sin embargo, que la diferencia en la cota de correlación obtenida al utilizar distintas semillas para la generación de parejas Golay incorreladas no es considerable. Asimismo, el número y posición de las interferencias en la IW son los mismos para todas las semillas evaluadas. De cualquier manera, se proporciona un listado con las semillas de los pares Golay iniciales y las μ secuencias LS de una familia de K secuencias, que permiten optimizar al máximo la cota θ dentro de la IW. Para cada caso, se proporciona un único valor de semilla $Wd_N^{(1}$ para la generación de uno de los pares. Al estar el otro par incorrelado con el anterior, su semilla de generación será $(Wd_N^{(1} + L_0/2) \bmod L_0$. Por otro lado, las μ secuencias concretas a las que corresponden las menores cotas θ en la IW se especifican según el orden en el que aparecen en el conjunto de K secuencias $\{G = g_k[l]; 0 \leq k \leq K-1; 0 \leq l \leq L-1\}$; en donde cada secuencia se obtiene a partir de (2.26). Se incluye además el porcentaje de interferencias en la IW para el caso de la ACF ($\%\frac{Int.IW_{ACF}}{IW}$) y la CCF ($\%\frac{Int.IW_{CCF}}{IW}$); evidentemente, un porcentaje inferior al 100 % implica la aparición de ceros en la IW. Calcular a partir de estos datos el porcentaje de interferencias respecto a la totalidad del código es inmediato sabiendo que el tamaño de la IW es $L - W - 1$.

Con esto, las tablas 3.11 a 3.14 recogen las principales características de la zona de interferencias de secuencias LS generadas a partir de pares Golay.

$\mathbf{M = 2; K = 2}$

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
4	11	2	0.375	0.25	0.375	0	0, 1	85.71 %	85.71 %
8	23	2	0.375	0.1875	0.375	0	0, 1	80 %	80 %
16	47	2	0.2188	0.1875	0.2188	0	0, 1	77.42 %	77.42 %
32	95	2	0.2188	0.1094	0.2188	0	0, 1	76.19 %	76.19 %
64	191	2	0.1719	0.1094	0.1719	0	0, 1	75.59 %	75.59 %
128	383	2	0.1328	0.0859	0.1328	12	0, 1	75.29 %	75.29 %
256	767	2	0.0977	0.082	0.0977	24	0, 1	75.15 %	75.15 %

Continuación de la Tabla 3.11

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
512	1535	2	0.0879	0.0566	0.0879	52	0, 1	75.07 %	75.07 %
1024	3071	2	0.0693	0.0459	0.0693	84	0, 1	75.04 %	75.04 %

Tabla 3.11: Estudio de la zona con interferencias de familias con $K = 2$ códigos LS generados a partir de 2-CSS según [SBH01].

$\mathbf{M = 2; K = 4}$

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
4	19	2	0.375	0.1875	0.375	0	0, 1	80 %	80 %
		3	0.5625	0.1875	0.5625	0	0, 1, 2		
		4	0.5625	0.1875	0.5625	0	0, 1, 2, 3		
8	39	2	0.2188	0.1875	0.2188	0	0, 1	77.42 %	77.42 %
		3	0.5313	0.1875	0.5313	0	0, 1, 2		
		4	0.5313	0.1875	0.5313	0	0, 1, 2, 3		
16	79	2	0.2188	0.1094	0.2188	0	0, 1	76.19 %	76.19 %
		3	0.5156	0.1094	0.5156	0	0, 1, 2		
		4	0.5156	0.1094	0.5156	0	0, 1, 2, 3		
32	159	2	0.1719	0.1094	0.1719	0	2, 3	75.59 %	75.59 %
		3	0.5078	0.1094	0.5078	0	0, 1, 2		
		4	0.5078	0.1094	0.5078	0	0, 1, 2, 3		
64	319	2	0.1328	0.0859	0.1328	12	0, 1	75.29 %	75.29 %
		3	0.5039	0.0742	0.5039	8	0, 1, 2		
		4	0.5039	0.0742	0.5039	8	0, 1, 2, 3		
128	639	2	0.0977	0.082	0.0977	24	0, 1	75.15 %	75.15 %
		3	0.502	0.0664	0.502	12	0, 1, 2		
		4	0.502	0.0664	0.502	12	0, 1, 2, 3		
256	1279	2	0.0879	0.0566	0.0879	52	0, 1	75.07 %	75.07 %
		3	0.501	0.0488	0.501	24	0, 1, 2		
		4	0.501	0.0488	0.501	24	0, 1, 2, 3		
512	2559	2	0.0693	0.0459	0.0693	84	0, 1	75.04 %	75.04 %
		3	0.5005	0.0405	0.5005	240	0, 1, 2		
		4	0.5005	0.0405	0.5005	240	0, 1, 2, 3		

Tabla 3.12: Estudio de la zona con interferencias de familias con $K = 4$ códigos LS generados a partir de 2-CSS según [SBH01].

$\mathbf{M = 2; K = 8}$

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
		2	0.5312	0.5312	0.1875	1	4, 7		
		3	0.5313	0.5313	0.3125	0	0, 5, 7		
		4	0.5313	0.5313	0.4375	0	0, 2, 5, 7		
4	35	5	0.5313	0.5313	0.5313	0	0, 1, 2, 3, 4	77.42 %	77.42 %
		6	0.5313	0.5313	0.5313	0	0, 1, 2, 3, 4, 6		
		7	0.5938	0.5313	0.5938	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5938	0.5313	0.5938	0	$0, \cdots, 7$		
		2	0.5156	0.5156	0.1563	2	0, 3		
		3	0.5156	0.5156	0.2656	0	0, 2, 7		
		4	0.5156	0.5156	0.2969	0	0, 2, 5, 7		
8	71	5	0.5156	0.5156	0.5156	0	0, 1, 2, 3, 4	76.19 %	76.19 %
		6	0.5156	0.5156	0.5156	0	0, 1, 2, 3, 4, 6		
		7	0.5469	0.5156	0.5469	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5469	0.5156	0.5479	0	$0, \cdots, 7$		
		2	0.5078	0.5078	0.1406	0	0, 3		
		3	0.5078	0.5078	0.2578	0	0, 2, 7		
		4	0.5078	0.5078	0.2734	0	0, 2, 5, 7		
16	143	5	0.5078	0.5078	0.5078	0	0, 1, 2, 3, 4	75.59 %	75.59 %
		6	0.5078	0.5078	0.5078	0	0, 1, 2, 3, 4, 6		
		7	0.5234	0.5078	0.5234	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5234	0.5078	0.5234	0	$0, \cdots, 7$		
		2	0.5039	0.5039	0.0859	0	0, 3		
		3	0.5039	0.5039	0.2539	0	0, 2, 7		
		4	0.5039	0.5039	0.2617	0	0, 2, 5, 7		
32	287	5	0.5039	0.5039	0.5039	0	0, 1, 2, 3, 4	75.29 %	75.29 %
		6	0.5039	0.5039	0.5039	0	0, 1, 2, 3, 4, 6		
		7	0.5117	0.5039	0.5117	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5117	0.5039	0.5117	0	$0, \cdots, 7$		
		2	0.502	0.502	0.082	4	0, 3		
		3	0.502	0.502	0.252	0	0, 2, 7		
		4	0.502	0.502	0.2559	0	0, 2, 5, 7		
64	575	5	0.502	0.502	0.502	0	0, 1, 2, 3, 4	75.15 %	75.15 %
		6	0.502	0.502	0.502	0	0, 1, 2, 3, 4, 6		
		7	0.5059	0.502	0.5059	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5059	0.502	0.5059	0	$0, \cdots, 7$		
		2	0.501	0.501	0.0645	8	0, 3		
		3	0.501	0.501	0.251	0	0, 2, 7		
		4	0.501	0.501	0.2529	0	0, 2, 5, 7		
128	1151	5	0.501	0.501	0.501	0	0, 1, 2, 3, 4	75.07 %	75.07 %
		6	0.501	0.501	0.501	0	0, 1, 2, 3, 4, 6		

Continuación de la Tabla 3.13

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
		7	0.5015	0.501	0.5029	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5015	0.501	0.5029	0	0, $\cdots$, 7		
256	2303	2	0.5005	0.5005	0.0557	20	0, 3	75.04 %	75.04 %
		3	0.5005	0.5005	0.2505	0	0, 2, 7		
		4	0.5005	0.5005	0.2515	0	0, 2, 5, 7		
		5	0.5005	0.5005	0.5005	0	0, 1, 2, 3, 4		
		6	0.5005	0.5005	0.5005	0	0, 1, 2, 3, 4, 6		
		7	0.5015	0.5005	0.5015	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5015	0.5005	0.5015	0	0, $\cdots$, 7		

Tabla 3.13: Estudio de la zona con interferencias de familias con $K = 8$ códigos LS generados a partir de 2-CSS según [SBH01].

$\mathbf{M = 2; K = 16}$

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
4	67	2	0.4844	0.4844	0.2188	0	3, 6	76.19 %	76.19 %
		3	0.5156	0.5156	0.4219	0	3, 7, 10		
		4	0.5156	0.5156	0.4531	0	3, 7, 10, 14		
		5	0.7656	0.7656	0.2969	0	0, 1, 5, 10, 14		
		6	0.7344	0.7344	0.4531	0	2, 5, 6, 8, 11, 15		
		7	0.7656	0.7656	0.4531	0	0, 1, 2, 5, 6, 11, 15		
		8	0.7656	0.7656	0.4688	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7656	0.7656	0.5156	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.7656	0.7656	0.7344	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7656	0.7656	0.7656	0	0, $\cdots$, 15		
8	135	2	0.4922	0.4922	0.1719	2	2, 7	75.59 %	75.59 %
		3	0.5078	0.5078	0.3984	0	3, 7, 10		
		4	0.5078	0.5078	0.4141	0	3, 7, 10, 14		
		5	0.7578	0.7578	0.2578	0	0, 1, 5, 10, 14		
		6	0.7578	0.7578	0.2891	0	2, 7, 8, 9, 12, 13		
		7	0.7578	0.7578	0.3984	0	0, 3, 4, 7, 9, 10, 13		
		8	0.7578	0.7578	0.4141	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7578	0.7578	0.5078	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		

Continuación de la Tabla 3.14

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
		14	0.7578	0.7578	0.7422	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7578	0.7578	0.7578	0	0, $\cdots$, 15		
16	271	2	0.4961	0.4961	0.1641	0	3, 6	75.29 %	75.29 %
		3	0.5039	0.5039	0.3867	0	3, 7, 10		
		4	0.5039	0.5039	0.3945	0	3, 7, 10, 14		
		5	0.7539	0.7539	0.2539	0	0, 1, 5, 10, 14		
		6	0.7539	0.7539	0.2695	0	2, 7, 8, 9, 12, 13		
		7	0.7539	0.7539	0.3867	0	0, 3, 4, 7, 9, 10, 13		
		8	0.7539	0.7539	0.3945	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7539	0.7539	0.5039	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.7539	0.7539	0.7461	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7539	0.7539	0.7539	0	0, $\cdots$, 15		
32	543	2	0.498	0.0957	0.498	0	3, 6	75.15 %	75.15 %
		3	0.502	0.502	0.3809	0	3, 7, 10		
		4	0.502	0.502	0.3848	0	3, 7, 10, 14		
		5	0.752	0.752	0.252	0	0, 1, 5, 10, 14		
		6	0.752	0.752	0.2598	0	2, 7, 8, 9, 12, 13		
		7	0.752	0.752	0.3809	0	0, 3, 4, 7, 9, 10, 13		
		8	0.752	0.752	0.3848	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.752	0.752	0.502	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.752	0.752	0.748	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.752	0.752	0.752	0	0, $\cdots$, 15		
64	1087	2	0.499	0.0957	0.499	4	2, 7	75.07 %	75.07 %
		3	0.501	0.501	0.3779	0	3, 7, 10		
		4	0.501	0.501	0.3799	0	3, 7, 10, 14		
		5	0.751	0.751	0.251	0	0, 1, 5, 10, 14		
		6	0.751	0.751	0.2549	0	2, 7, 8, 9, 12, 13		
		7	0.751	0.751	0.3779	0	0, 3, 4, 7, 9, 10, 13		
		8	0.751	0.751	0.3799	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.751	0.751	0.501	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		

Continuación de la Tabla 3.14

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
		14	0.751	0.751	0.749	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.751	0.751	0.751	0	0, $\cdots$, 15		

Tabla 3.14: Estudio de la zona con interferencias de familias con $K = 16$ códigos LS generados a partir de 2-CSS según [SBH01].

Independientemente de las parejas Golay elegidas para formar las secuencias LS, dentro de la IW de los K códigos de una familia de longitud L aparecen el mismo número de interferencias o valores no nulos, situados además en las mismas posiciones. Sin embargo, la magnitud y signo de las interferencias sí son distintos para cada código LS, de modo que en función del número μ de usuarios simultáneos pueden elegirse los de menores valores de cota de correlación. En las tablas anteriores puede observarse como en la IW aparecen valores nulos, en concreto la IW consiste aproximadamente en un 76.35 % de interferencias (con una desviación estándar del 2 %) y un 23.65 % de ceros; para longitudes largas el número de interferencias se reduce al 75 %. En la figura 3.12 se representa el porcentaje de interferencias en la longitud total de la secuencia LS para distintos valores de K. Puede observarse que cuanto mayor es el número de códigos K disponibles en una familia mayor es el porcentaje de interferencias en la secuencia; ahora bien, también es mayor el valor del pico principal de la ACF respecto a la longitud de la secuencia[3], como puede verse en la figura 3.13.

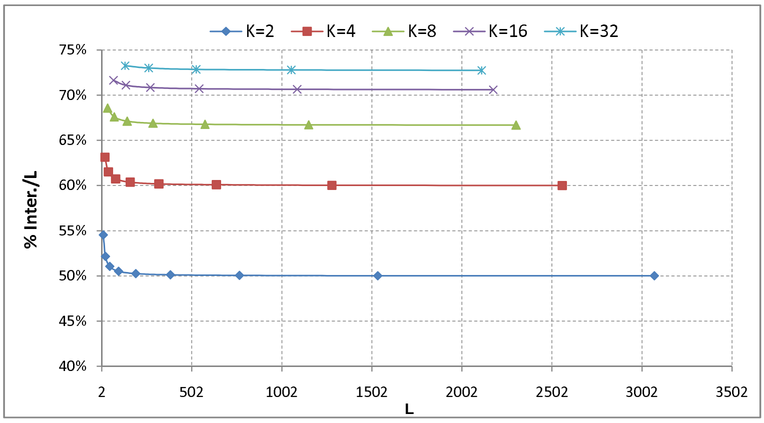


Figura 3.12: Porcentaje de interferencias en secuencias LS generadas según [SBH01], en función del número máximo de usuarios simultáneos K y la longitud L.

[3]Las secuencias LS están compuestas por valores $\{-1, 0, 1\}$, de modo que no todos los bits de la secuencia contribuyen a incrementar la magnitud del lóbulo principal de ACF, esto es, $R_{g_k,g_k}(0) = C_{g_k,g_k}(0) < L$.

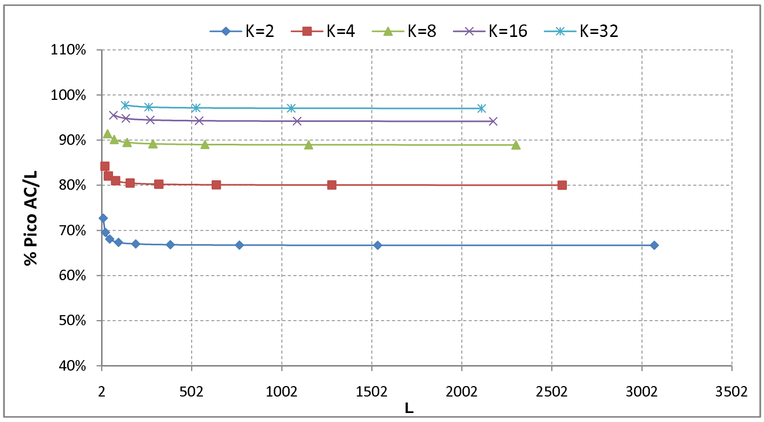
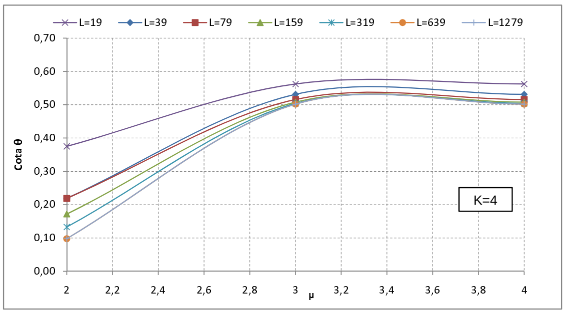
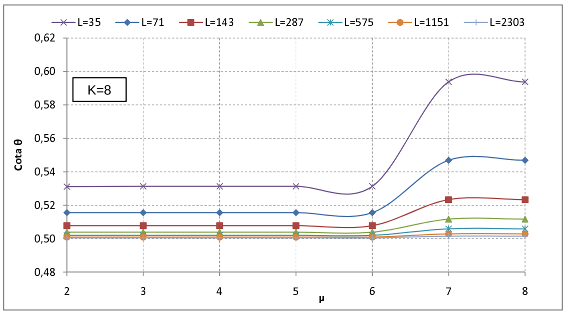


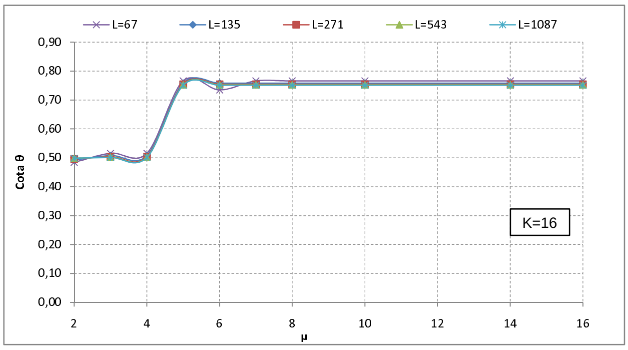
Figura 3.13: Valor del pico principal de la ACF de secuencias LS [SBH01] respecto a la longitud L del código.

Ya se adelantaba anteriormente que las interferencias fuera de la IFW podían llegar a ser elevadas, nótese que para un número de usuarios $\mu \geq 3$ la cota θ obtenida no baja de 0.5. Es decir, con tres emisores simultáneos en el mejor de los casos las interferencias en la IW serían del orden de la mitad del pico principal de ACF. Cuando el número de usuarios simultáneos $\mu = 2$ y el número de códigos de la familia es $K = 2$ ó $K = 4$ puede apreciarse una disminución significativa de la cota θ al aumentar la longitud de las secuencias (para K=4 véase la figura 3.14.a). Sin embargo, en el resto de casos, la diferencia no supera el 10 % del valor del pico de auto-correlación, según se observa en la figura 3.14.b y 3.14.c. Por otro lado, para la misma longitud de secuencia, incrementar el número de emisores simultáneos por encima de un determinado valor, supone también un aumento de la cota θ; a excepción de cuando $K = 8$, en donde el valor máximo de las interferencias aún cuando aumenta μ es alrededor de la mitad del pico principal de la ACF (para $L = 35$ la diferencia entre la cota de correlación asociada a $\mu = 2$ usuarios simultáneos y la cota asociada a $\mu = 8$ es de 0.0626, siendo menor todavía para longitudes más elevadas). En general, para códigos de longitud L similar y el mismo número μ de usuarios simultáneos, escoger la familia con menor número de códigos K (con $\mu \leq K$) supone, cuanto menos, una cota θ similar a la conseguida con K mayores y, para determinados casos, valores menores de θ en la IW, como se ilustra en la figura 3.15.



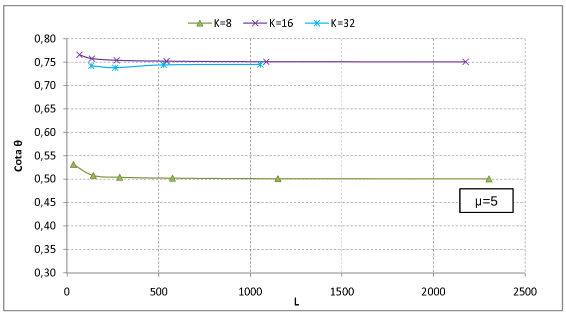


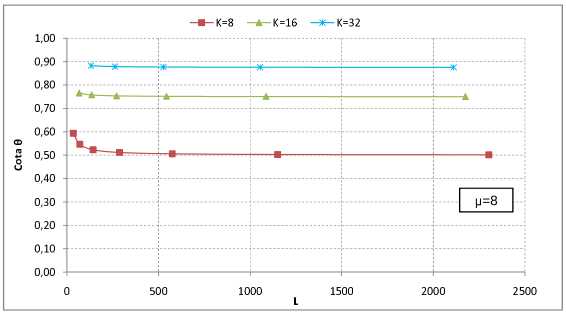
(a) $K = 4$ (b) $K = 8$



(c) $K = 16$

Figura 3.14: Valores de cota θ en la IW para códigos LS [SBH01], en función del número de códigos K de la familia y del número μ de usuarios simultáneos.





(a) $\mu = 5$ (b) $\mu = 8$

Figura 3.15: Valores de cota θ en la IW para códigos LS [SBH01], para el caso de a) $\mu = 5$ y b) $\mu = 8$ usuarios simultáneos, en función del número de códigos K de la familia y de la longitud L de las secuencias.

3.4.2. Estudio de la IW asociada a códigos LS generados a partir de CSS incorrelados

La tabla 3.15 recoge los resultados de cota de correlación y número de interferencias en la IW cuando los códigos LS se generan a partir de CSS y $K = 16$. Para $K = 4$, esto es $M = 2$, los resultados son idénticos a los mostrados en la tabla 3.12 anterior. Al igual que en las tablas asociadas a códigos LS generados a partir de Golay, se muestra aquí una única semilla $Wd_N^{(m}$ para la generación de uno de los CSS que componen el código LS; los $M-1$ CSS restantes son incorrelados al anterior, obteniéndose sus semillas como: $(Wd_N^{(m} + i \cdot \frac{L}{M}) \bmod L; 1 \leq i \leq M-1$.

Con este método de generación el número K de códigos de una familia crece más rápidamente que con el propuesto en [SBH01] ($K = M^2 = 2^{2m},\ m \in \mathbb{N}$ frente a $K = 2^n,\ n \in \mathbb{N}$), lo que se traduce en una menor variedad de longitudes de código.

M = 4; K = 16

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
4	73	2	0.2656	0.2656	0.1094	0	1, 2	89.96 %	89.96 %
		3	0.2656	0.2656	0.25	1	0, 3, 5		
		4	0.2969	0.2656	0.2969	0	8, 9, 11, 15		
		5	0.5	0.2656	0.5	0	0, 1, 3, 7, 15		
		6	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9		
		7	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9, 10		
		8	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9, 10, 11		
		10	0.7344	0.2656	0.7344	0	0, 3, 5, 7, 8, 9, 10, 11, 13, 15		
		14	0.7656	0.3125	0.7656	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		
		16	0.7656	0.3125	0.7656	0	0, $\cdots$, 15		
16	301	2	0.1953	0.1953	0.1094	0	1, 2	80.70 %	78.60 %
		3	0.2617	0.1797	0.2617	1	3, 5, 7		
		4	0.25	0.1953	0.25	0	1, 3, 8, 10		
		5	0.4805	0.1953	0.4805	1	7, 8, 9, 11, 15		
		6	0.5039	0.1953	0.5039	0	0, 1, 2, 3, 8, 9		
		7	0.5039	0.1953	0.5039	0	0, 1, 2, 3, 8, 9, 10		
		8	0.5039	0.1953	0.5039	1	0, 1, 2, 3, 8, 9, 10, 11		
		10	0.7641	0.1953	0.7461	0	0, 1, 2, 3, 4, 6, 9, 12, 14		
		14	0.7539	0.1953	0.7539	1	0, 1, 2, 3, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15		
		16	0.7539	0.2266	0.7539	0	0, $\cdots$, 15		
		2	0.1875	0.1875	0.0675	12	9, 11		
		3	0.1855	0.1367	0.1855	4	4, 5, 7		
		4	0.1973	0.1973	0.1367	4	4, 5, 6, 7		
		5	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 13		

Continuación de la Tabla 3.15

L_0	L	μ	θ	θ_{AC}	θ_{CC}	semilla	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
64	1213	6	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 13, 15	79.90 %	81.20 %
		7	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 12, 13, 14		
		8	0.749	0.1875	0.749	0	4, 5, 6, 7, 12, 13, 14, 15		
		10	0.749	0.1875	0.749	0	0, 1, 6, 7, 8, 9 10, 11, 14, 15		
		14	0.751	0.1875	0.751	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		
		16	0.751	0.1875	0.751	0	0, $\cdots$, 15		
256	4861	2	0.0637	0.0637	0.0637	8	5, 7	78.61 %	77.83 %
		3	0.1875	0.0649	0.1875	0	4, 5, 6		
		4	0.1875	0.0649	0.1875	0	4, 5, 6, 7		
		5	0.5017	0.0649	0.5017	0	4, 5, 6, 7, 13		
		6	0.5017	0.0649	0.5017	0	4, 5, 6, 7, 13, 15		
		7	0.5022	0.0649	0.5022	0	4, 5, 6, 7, 12, 13, 14		
		8	0.5022	0.0649	0.5022	0	4, 5, 6, 7, 12, 13, 14, 15		
		10	0.7498	0.105	0.7498	0	4, 5, 6, 7, 10, 11, 12, 13, 14, 15		
		14	0.7502	0.1099	0.7502	0	0, 1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15		
		16	0.7502	0.1108	0.7502	0	0, $\cdots$, 15		

Tabla 3.15: Estudio de la zona con interferencias de familias con $K = 16$ códigos LS generados a partir de 4-CSS según [ZYH05].

En primer lugar, puede observarse como se obtienen menores valores nulos en la IW en comparación con el método de generación a partir de pares Golay. El porcentaje de interferencias en la longitud total de la secuencia para distintos valores de K se recoge en la figura 3.16. Aquí puede observarse que, para la misma longitud de secuencia, cuanto mayor sea el número de códigos disponibles mayor será el número de interferencias en la IW. El número y lugar en el que aparecen las interferencias dependen de si se trata de la ACF o la CCF, aunque para ambos casos los valores obtenidos son los mismos para todos los códigos de la familia. El porcentaje de interferencias que aparece en la figura 3.16 se ha calculado considerando el mayor de los valores obtenidos en las funciones de correlación: max(nº interferencias ACF, nº interferencias CCF). Por otro lado, con este método se inserta un mayor número de ceros entre secuencias de los CSS, y por tanto es menor la relación pico principal de auto-correlación respecto a la longitud de la secuencia en comparación con el método descrito en el apartado anterior.

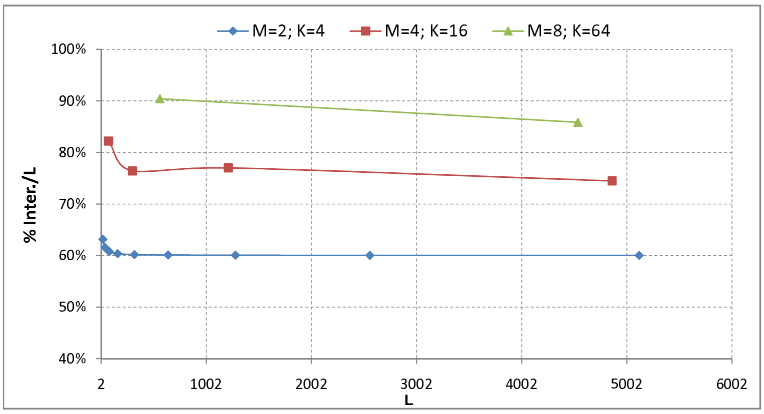


Figura 3.16: Porcentaje de interferencias en secuencias LS generadas según [ZYH05], en función del número máximo de usuarios simultáneos K y la longitud L.

Sin embargo, los valores de cota θ en la IW son menores (para $K > 4$) con este método de generación que con el basado en pares Golay, como puede observarse en la figura 3.17. Se muestran aquí en línea continua los valores obtenidos para $K = 16$ con códigos LS generados a partir de CSS, y en línea discontinua los obtenidos a partir de pares Golay. Cuando $M = 4$ y el número de usuarios simultáneos es reducido ($\mu \leq 4$ para $K = 16$) utilizar secuencias de mayor longitud conlleva menores valores de θ; ahora bien, estas diferencias dejan de ser notables conforme aumenta el número de usuarios. Finalmente, en la figura 3.18 se representa la cota θ obtenida para distintas longitudes de secuencia y número M de conjuntos incorrelados. Se observa también aquí que la cota θ se reduce cuando se usan conjuntos con más de dos secuencias complementarias.

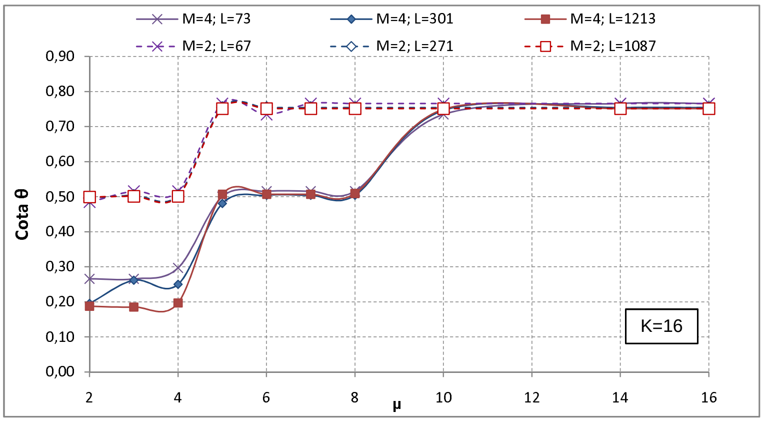


Figura 3.17: Valores de cota θ en la IW para familias con $K = 16$ códigos LS, en el caso de usar el algoritmo de generación propuesto en [ZYH05] (línea continua) o el propuesto en [SBH01] (línea discontinua).

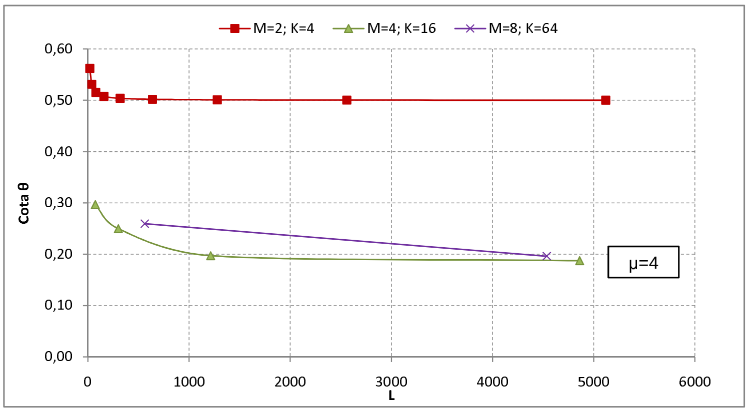


Figura 3.18: Valores de cota θ en la IW para $\mu = 4$ usuarios simultáneos y códigos LS generados según [ZYH05].

3.4.3. Influencia de la recepción parcial del código LS en la cota de auto-correlación

La pérdida de bits al inicio de la secuencia implica no sólo un aumento de los lóbulos laterales de auto-correlación en la IW de códigos LS, sino también la aparición de interferencias en la IFW, como puede verse en la figura 3.19. Se representa en esta figura la ACF correspondiente a la recepción de todo el código $LS(2, 16, 4)$ emitido, y la ACF obtenida cuando se reciben los últimos 27 bits (el 35 % de la señal). En este último caso la magnitud del pico principal es menor y aparecen interferencias en la IFW, incrementándose además el valor de las ya existentes en la IW; aún así continua siendo posible distinguir sin problemas el pico principal.

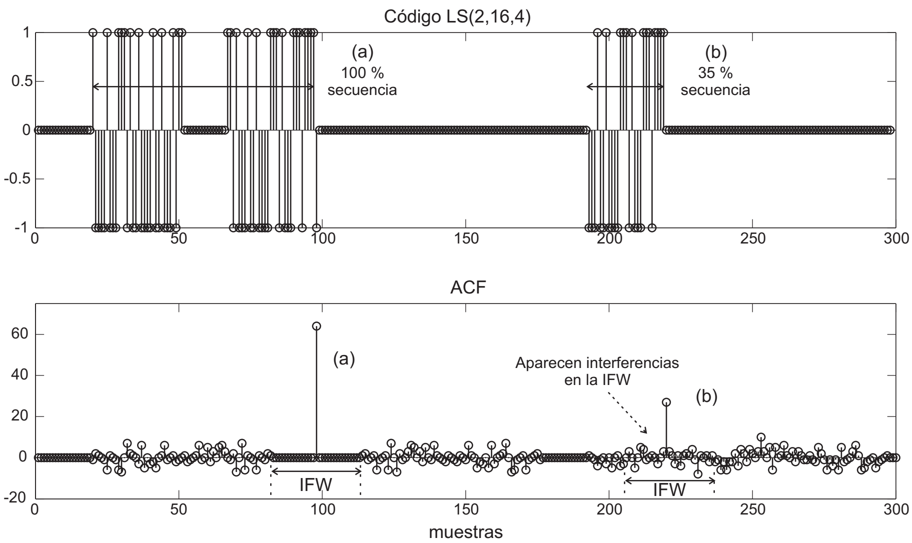


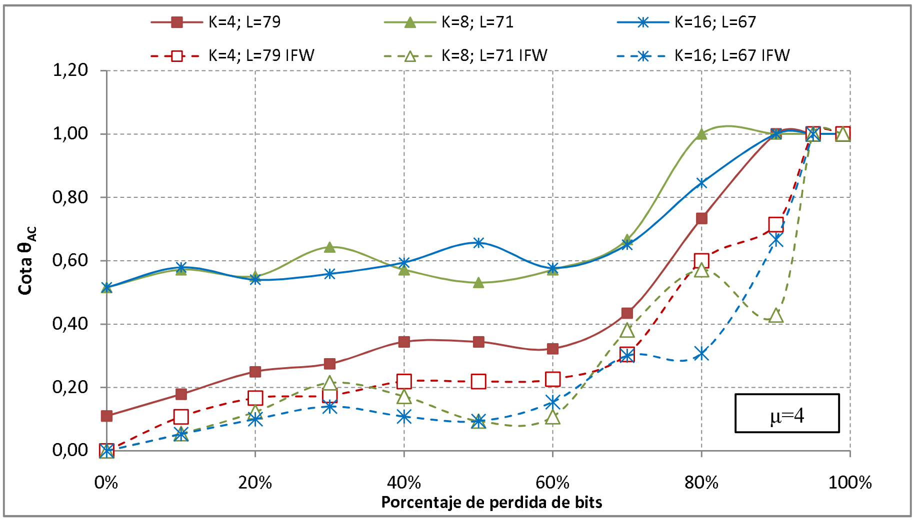
Figura 3.19: Ejemplo de ACF de un código $LS(2, 16, 4)$ cuando a) se recibe todo el código y b) se pierde el inicio del código (sólo se considera el 35 %).

La figura 3.20 muestra en línea discontinua los valores de cota θ_{AC} obtenidos en lo que debería ser la IFW, y en línea continua los obtenidos en la IW según se pierde un mayor número de bits al inicio de la secuencia. En este ejemplo los códigos LS se han generado con el método propuesto en [SBH01] y se han elegido los $\mu = 4$ códigos LS con interferencias de menor valor en la IW. En el caso (a) se muestran los resultados de códigos con longitud aproximada 70 bits; y en el (b) con longitud 1170 bits (la longitud es función del número de códigos K disponibles en la familia). El valor de cota θ_{AC} en la IFW se ha calculado considerando una zona alrededor del pico principal de tamaño igual a la IFW obtenida en condiciones ideales (recibiendo el 100 % del código). Podría ocurrir que la cota en la IW sea $\theta_{AC} = 1$ y la cota en la IFW sea baja; esto correspondería al caso de tener un pico principal de la misma magnitud que ciertos lóbulos laterales, pero rodeado por una zona de mayor tamaño que la IFW con interferencias de menor valor. En estas circunstancias, sólo si existe sincronismo podría detectarse el eco emitido; en caso contrario, no sería posible distinguir qué pico es el principal. Para secuencias cortas (caso a) la cota θ_{AC} dentro de la IFW permanece por debajo del 0.5 para porcentajes de pérdida de hasta el 75 %. Con secuencias de mayor longitud también es posible reducir el tamaño de la zona ciega; así, en el caso b el pico principal se distingue con claridad de los laterales que aparecen en la IFW incluso cuando se pierde el 95 % del código. Cuando el número de usuarios aumenta el comportamiento es parecido, a modo de ejemplo se muestra en la figura 3.21 el valor de la cota θ_{AC} dentro y fuera de la IFW cuando se trabaja con $\mu = 8$ usuarios a la vez.

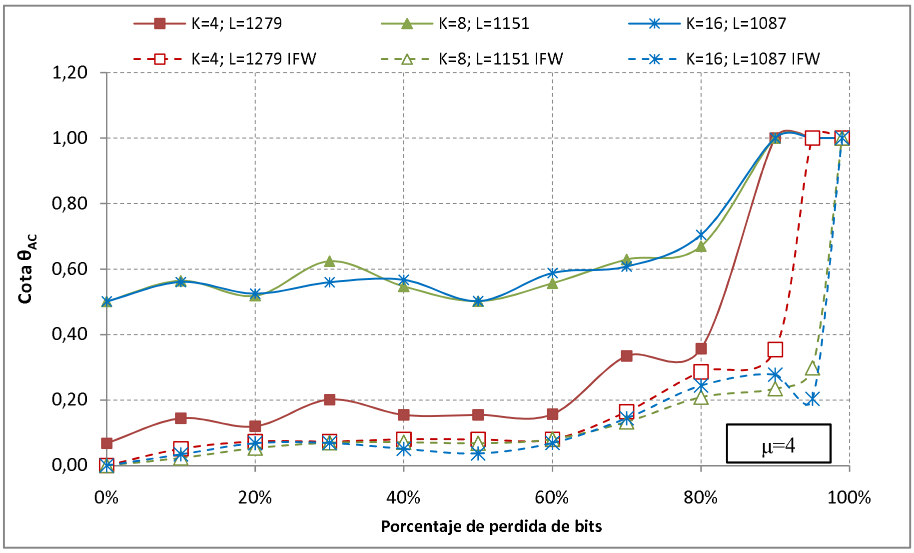
El estudio descrito anteriormente se ha repetido para el caso de trabajar con códigos LS generados a partir de CSS según [ZYH05], los resultados obtenidos con $\mu = 4$ códigos emitidos simultáneamente se muestran en la figura 3.22. Los valores de la cota θ_{AC} dentro de la IFW son similares a los conseguidos con el método de generación basado en pares Golay cuando el número de códigos K de la familia es el mismo; aunque se obtienen valores menores de θ_{AC} fuera de la IFW.

3.5. Códigos T-ZCZ

Las parejas T-ZCZ surgen tras la búsqueda de códigos ortogonales generalizados que en emisión periódica no requieran insertar zonas de guarda entre emisiones para evitar interferencias alrededor del origen. En emisión aperiódica, la suma de las funciones de correlación de las dos secuencias del par presenta zonas de correlación cero no sólo alrededor del origen sino también en los dos extremos laterales. Se lleva a cabo aquí un estudio de la zona en la que las interferencias están confinadas (IW), teniendo en cuenta que su tamaño y la cota de correlación aperiódica θ asociada depende del mecanismo de generación utilizado (véase la sección 2.2.7).

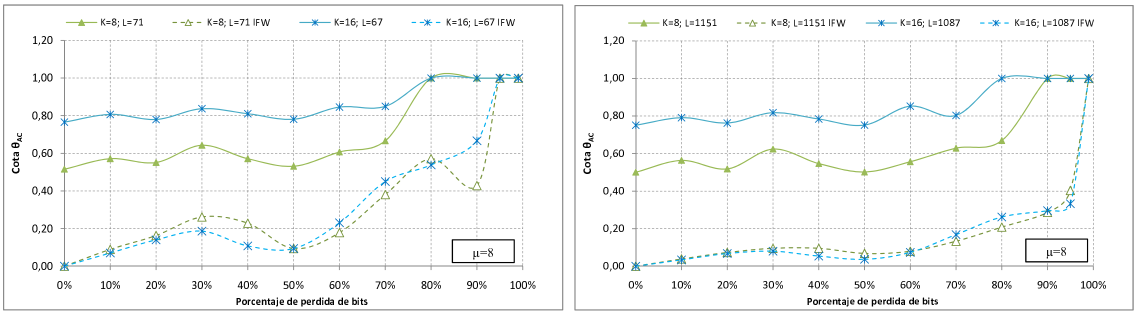


(a) $67 \leq L \leq 79$



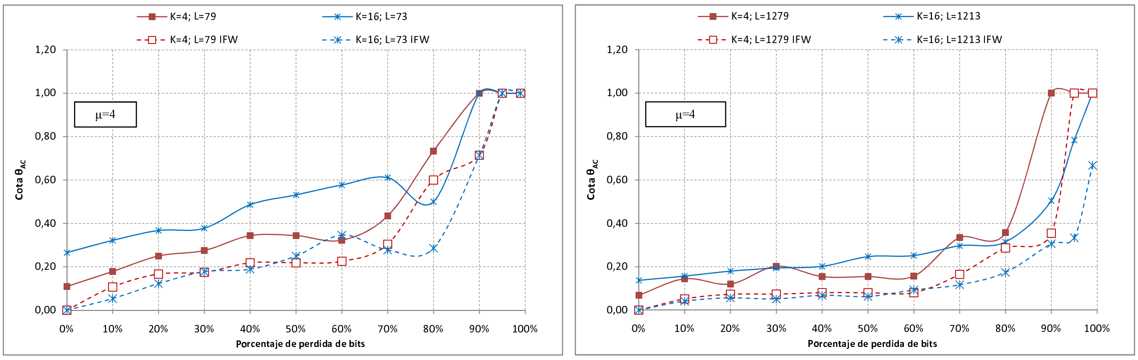
(b) $1087 \leq L \leq 1279$

Figura 3.20: θ_{AC} en función del porcentaje de secuencia perdido para $\mu = 4$ secuencias LS [SBH01] de longitud similar.



(a) $L = 67$ y $L = 71$ (b) $L = 1087$ y $L = 1151$

Figura 3.21: θ_{AC} en función del porcentaje de secuencia perdido para $\mu = 8$ secuencias LS [SBH01] de longitud similar.



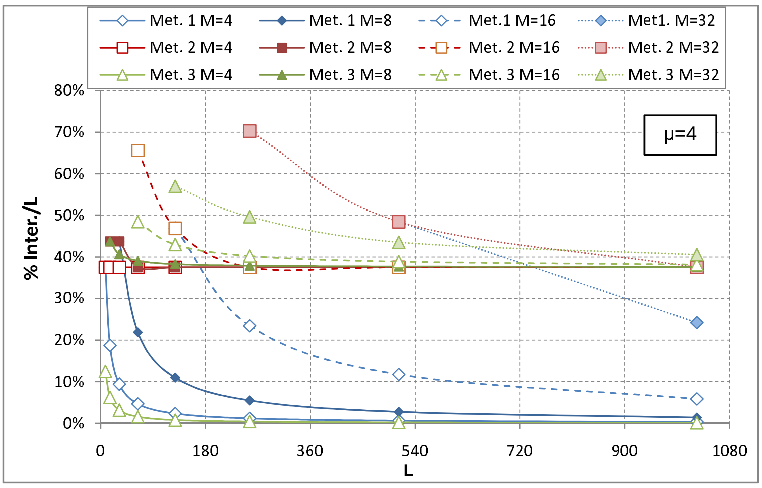
(a) $L = 73$ y $L = 79$ (b) $L = 1213$ y $L = 1279$

Figura 3.22: θ_{AC} en función del porcentaje de secuencia perdido para $\mu = 4$ secuencias LS [ZYH05] de longitud similar.

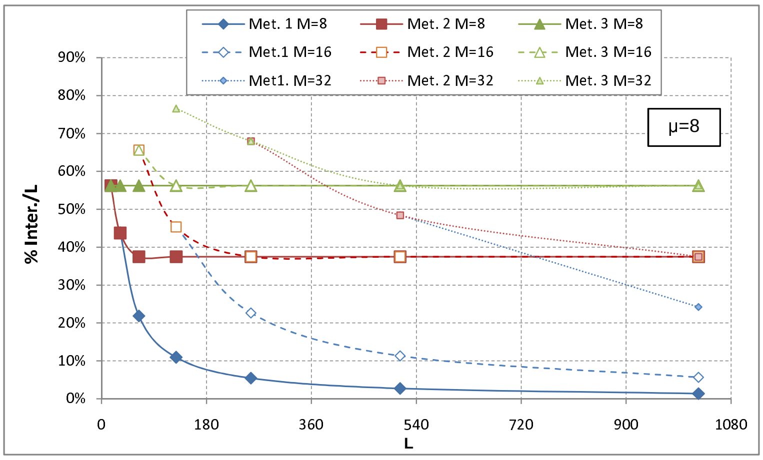
3.5.1. Estudio de la zona con interferencias

Para cada método se ha evaluado el número y magnitud de las interferencias obtenidas en la IW en función del número M de códigos de la familia y la longitud L de los mismos. En la SACF el número y posición de los lóbulos laterales coinciden en los M códigos, aunque no su magnitud. Sin embargo, el número de interferencias en la SCCF depende de los códigos considerados, de modo que es posible elegir las μ combinaciones que menor número de interferencias presenten. En cualquier caso, en un sistema práctico, interesa más reducir la magnitud de las interferencias que su número (una única interferencia de valor elevado puede confundirse con el pico principal de auto-correlación). Por tanto, se ha establecido como primer criterio de búsqueda elegir las μ parejas T-ZCZ con cota θ de menor valor en la IW, reduciendo así el impacto de las ISI y MAI cuando algún eco se recibe fuera de las IFWs. Como segundo criterio se ha establecido elegir, de entre la selección previa, aquellas con menor número de interferencias.

Al igual que sucede con los códigos LS, en la IW también aparecen ceros. El método con la IW más estrecha y con menor número de interferencias en dicha IW es el primero; salvo cuando $M = 4$, que es el tercero. En la figura 3.23 se representa el porcentaje de interferencias en la longitud total de la secuencia para los tres métodos, y en el caso de a) $\mu = 4$ y b) $\mu = 8$ usuarios simultáneos. Puede observarse como, para el primer método, el porcentaje de interferencias disminuye drásticamente conforme aumenta la longitud de la secuencia. El segundo y tercer método presentan un número mayor de interferencias que el primero (a excepción de cuando $M = 4$ y $\mu \leq 4$ en donde el porcentaje de interferencias es algo menor en el tercer método).



(a) $\mu = 4$

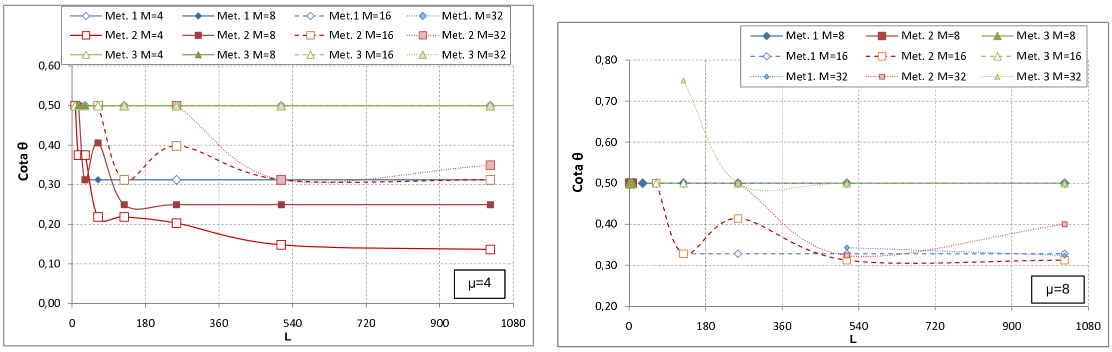


(b) $\mu = 8$

Figura 3.23: Porcentaje de interferencias en pares T-ZCZ, en función del método de generación, el número M de códigos de la familia y la longitud L de los mismos.

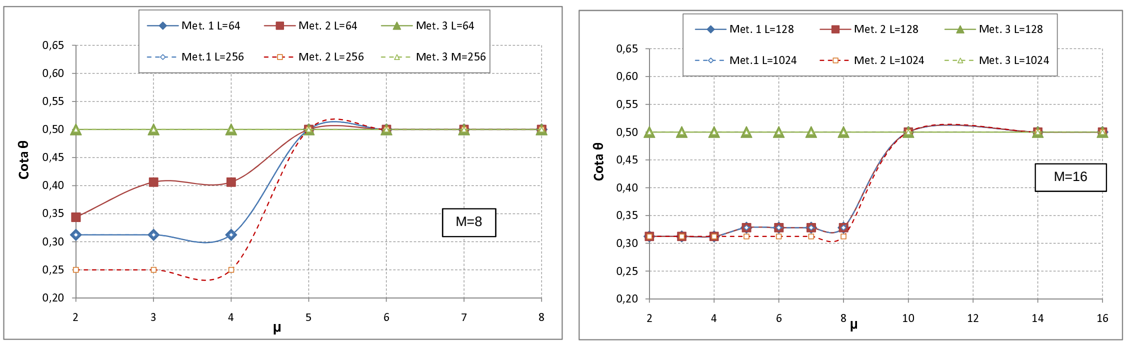
La magnitud de las interferencias dentro de la IW para el tercer método es, cuanto menos, del orden de la mitad del pico principal de la SACF, esto es, $\theta \geq 0.5$ para cualquier longitud y número de usuarios simultáneos. Con los otros dos métodos es posible obtener

$\theta < 0.5$ para determinadas combinaciones de M y μ, siendo algo menores los valores de θ conseguidos con el segundo método. En la figura 3.24 se representan los valores de cota θ en función de la longitud de la secuencia, para el caso de a) $\mu = 4$ y b) $\mu = 8$ usuarios simultáneos. La figura 3.25 muestra también los valores de cota θ, esta vez en función del número de usuarios simultáneos y para un número de códigos en la familia a) $M = 8$ y b) $M = 16$. Puede observarse en estas figuras, como cuando $M \leq 4$ ó $\mu \leq \frac{M}{2}$ los valores de cota de correlación obtenidos son menores de 0.5.



(a) $\mu = 4$ (b) $\mu = 8$

Figura 3.24: Valores de cota θ en la IW para códigos T-ZCZ, en función del método de generación, el número M de códigos de la familia y la longitud L de los mismos.



(a) $M = 8$ (b) $M = 16$

Figura 3.25: Valores de cota θ en la IW para códigos T-ZCZ, en función del método de generación y el número μ de códigos simultáneos.

Se muestra en las tablas 3.16 a 3.18 la selección de secuencias que permite reducir al máximo la magnitud de las interferencias para el caso del segundo método, ya que es el de menor cota θ en la IW. Asimismo, se especifica el porcentaje de ocupación de interferencias en dicha IW. Si se representase el número de interferencias respecto a la longitud total del código, se observaría que tiende a estabilizarse en torno al 37.50 % conforme aumenta la longitud de dicho código.

M = 4

L	W	μ	θ	θ_{AC}	θ_{CC}	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
16	4	2	0.375	0.25	0	0, 2	57.14 %	0 %
		3	0.375	0.375	0.375	0, 1, 2		85.71 %
		4	0.375	0.375	0.375	0, 1, 2, 3		85.71 %
32	8	2	0.1875	0.1875	0	0, 2	57.33 %	0 %
		3	0.375	0.1875	0.375	0, 1, 2		80 %
		4	0.375	0.1875	0.375	0, 1, 2, 3		80 %
64	16	2	0.2188	0.2188	0	0, 2	51.61 %	0 %
		3	0.2188	0.2188	0.2188	0, 1, 2		77.42 %
		4	0.2188	0.2188	0.2188	0, 1, 2, 3		77.42 %
128	32	2	0.2031	0.2031	0	0, 2	50.79 %	0 %
		3	0.2188	0.2031	0.2188	0, 1, 2		76.19 %
		4	0.2188	0.2031	0.2188	0, 1, 2, 3		76.19 %
256	64	2	0.1484	0.1484	0	0, 2	50.39 %	0 %
		3	0.2031	0.1484	0.2031	0, 1, 2		75.59 %
		4	0.2031	0.1484	0.2031	0, 1, 2, 3		75.59 %
512	128	2	0.1367	0.1367	0	0, 2	50.20 %	0 %
		3	0.1484	0.1367	0.1484	0, 1, 2		75.29 %
		4	0.1484	0.1367	0.1484	0, 1, 2, 3		75.29 %
1024	256	2	0.0801	0.0801	0	0, 2	50.10 %	0 %
		3	0.1367	0.0801	0.1367	0, 1, 2		75.15 %
		4	0.1367	0.0801	0.1367	0, 1, 2, 3		75.15 %

Tabla 3.16: Estudio de la zona con interferencias de familias con $M = 4$ códigos T-ZCZ generados según el método 2.

M = 8

L	W	μ	θ	θ_{AC}	θ_{CC}	códigos	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
16	2	2	0.5	0.5	0	0, 4	63.64 %	0 %
		3	0.5	0.5	0.25	0, 3, 4		54.55 %
		4	0.5	0.5	0.25	0, 3, 4, 7		54.55 %
		5	0.5	0.5	0.5	0, 1, 2, 3, 4		81.82 %
		6	0.5	0.5	0.5	0, 1, 2, 3, 4, 5		81.82 %
		7	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6		81.82 %
		8	0.5	0.5	0.5	0, 1, $\cdots$, 7		81.82 %
32	4	2	0.3125	0.3125	0	0, 4	43.38 %	0 %
		3	0.3125	0.3125	0.25	0, 3, 4		60.87 %
		4	0.3125	0.3125	0.25	0, 3, 4, 7		60.87 %
		5	0.5	0.3125	0.5	0, 1, 2, 3, 4		60.87 %
		6	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5		60.87 %

Continuación de la Tabla 3.17

L	W	μ	θ	θ_{AC}	θ_{CC}	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
		7	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6		60.87 %
		8	0.5	0.3125	0.5	0, 1, $\cdots$, 7		60.87 %
		2	0.3438	0.3438	0	0, 4		0 %
		3	0.4063	0.4063	0.2188	0, 3, 4		51.06 %
		4	0.4063	0.4063	0.2188	0, 3, 4, 7		51.06 %
64	8	5	0.5	0.4063	0.5	0, 1, 2, 3, 4	38.30 %	51.06 %
		6	0.5	0.4063	0.5	0, 1, 2, 3, 4, 5		51.06 %
		7	0.5	0.4063	0.5	0, 1, 2, 3, 4, 5, 6		51.06 %
		8	0.5	0.4063	0.5	0, 1, $\cdots$, 7		51.06 %
		2	0.25	0.25	0	0, 4		0 %
		3	0.25	0.25	0.2188	0, 3, 4		50.53 %
		4	0.25	0.25	0.2188	0, 3, 4, 7		50.53 %
128	16	5	0.5	0.25	0.5	0, 1, 2, 3, 4	35.79 %	50.53 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.53 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.53 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.53 %
		2	0.25	0.25	0	0, 4		0 %
		3	0.25	0.25	0.1406	0, 3, 4		50.26 %
		4	0.25	0.25	0.1406	0, 3, 4, 7		50.26 %
256	32	5	0.5	0.25	0.5	0, 1, 2, 3, 4	34.55 %	50.26 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.26 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.26 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.26 %
		2	0.25	0.25	0	0, 4		0 %
		3	0.25	0.25	0.1328	0, 3, 4		50.13 %
		4	0.25	0.25	0.1328	0, 3, 4, 7		50.13 %
512	64	5	0.5	0.25	0.5	0, 1, 2, 3, 4	33.94 %	50.13 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.13 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.13 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.13 %
		2	0.25	0.25	0	0, 4		0 %
		3	0.25	0.25	0.1211	0, 3, 4		50.13 %
		4	0.25	0.25	0.1211	0, 3, 4, 7		50.13 %
1024	128	5	0.5	0.25	0.5	0, 1, 2, 3, 4	33.64 %	50.13 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.13 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.13 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.13 %

Tabla 3.17: Estudio de la zona con interferencias de familias con $M = 8$ códigos T-ZCZ generados según el método 2.

M = 16

L	W	μ	θ	θ_{AC}	θ_{CC}	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
64	4	2	0.5	0.5	0	0, 8	56.36 %	0 %
		3	0.5	0.5	0.125	0, 7, 8		76.36 %
		4	0.5	0.5	0.125	0, 7, 8, 15		76.36 %
		5	0.5	0.5	0.25	0, 3, 5, 6, 8		76.36 %
		6	0.5	0.5	0.25	0, 3, 5, 6, 8, 11		76.36 %
		7	0.5	0.5	0.25	0, 3, 5, 6, 7, 11, 13		76.36 %
		8	0.5	0.5	0.25	0, 3, 5, 6, 8, 11, 13, 14		76.36 %
		10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		76.36 %
		14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		76.36 %
		16	0.5	0.5	0.5	0, 1, $\cdots$, 15		76.36 %
128	8	2	0.3125	0.3125	0	2, 10	37.84 %	0 %
		3	0.3125	0.3125	0.125	2, 5, 10		54.05 %
		4	0.3125	0.3125	0.125	2, 5, 10, 13		54.05 %
		5	0.3281	0.3281	0.25	0, 3, 5, 6, 8		52.25 %
		6	0.3281	0.3281	0.25	0, 3, 5, 6, 8, 11		52.25 %
		7	0.3281	0.3281	0.25	0, 3, 5, 6, 7, 11, 13		52.25 %
		8	0.3281	0.3281	0.25	0, 3, 5, 6, 8, 11, 13, 14		52.25 %
		10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		54.05 %
		14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		54.05 %
		16	0.5	0.5	0.5	0, 1, $\cdots$, 15		54.05 %
256	16	2	0.3359	0.3359	0	0, 8	33.18 %	0 %
		3	0.3984	0.3984	0.1172	0, 7, 8		43.05 %
		4	0.3984	0.3984	0.1172	0, 7, 8, 15		43.05 %
		5	0.3984	0.3984	0.25	0, 5, 6, 8, 13		43.05 %
		6	0.3984	0.3984	0.25	0, 5, 6, 8, 13, 14		43.05 %
		7	0.4141	0.4141	0.25	0, 3, 5, 6, 7, 11, 13		43.05 %
		8	0.4141	0.4141	0.25	0, 3, 5, 6, 8, 11, 13, 14		43.05 %
		10	0.5	0.4141	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		43.05 %
		14	0.5	0.4141	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		43.05 %
		16	0.5	0.4141	0.5	0, 1, $\cdots$, 15		43.05 %
512	32	2	0.3125	0.3125	0	0, 8	30.87 %	0 %
		3	0.3125	0.3125	0.3125	0, 7, 8		42.95 %
		4	0.3125	0.3125	0.3125	0, 7, 8, 15		42.95 %
		5	0.3125	0.3125	0.3125	0, 3, 5, 6, 8		42.95 %
		6	0.3125	0.3125	0.3125	0, 3, 5, 6, 8, 11		42.95 %

Continuación de la Tabla 3.18

L	W	μ	θ	θ_{AC}	θ_{CC}	códigos	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
		7	0.3125	0.3125	0.3125	0, 3, 5, 6, 7, 11, 13		42.95 %
		8	0.3125	0.3125	0.3125	0, 3, 5, 6, 8, 11, 13, 14		42.95 %
		10	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		42.95 %
		14	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		42.95 %
		16	0.5	0.3125	0.5	0, 1, $\cdots$, 15		42.95 %
1024	64	2	0.3125	0.3125	0	0, 8	29.72 %	0 %
		3	0.3125	0.3125	0.3125	0, 7, 8		42.91 %
		4	0.3125	0.3125	0.3125	0, 7, 8, 15		42.91 %
		5	0.3125	0.3125	0.3125	0, 1, 6, 7, 8		42.91 %
		6	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9		42.91 %
		7	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9, 14		42.91 %
		8	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9, 14, 15		42.91 %
		10	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		42.91 %
		14	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		42.91 %
		16	0.5	0.3125	0.5	0, 1, $\cdots$, 15		42.91 %

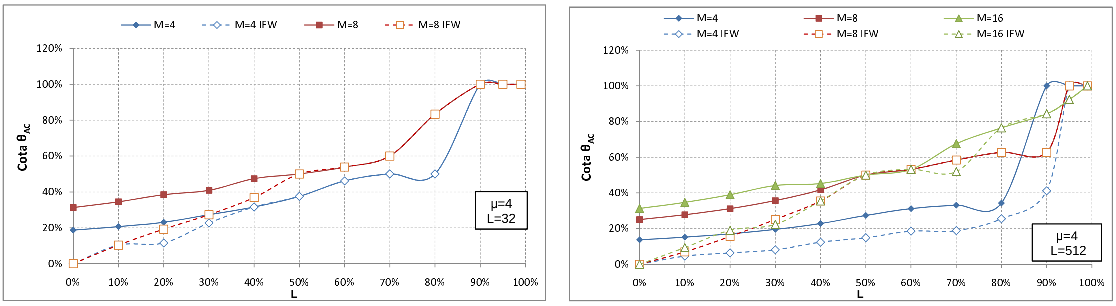
Tabla 3.18: Estudio de la zona con interferencias de familias con $M = 16$ códigos T-ZCZ generados según el método 2.

Los resultados mostrados anteriormente son válidos cuando las secuencias del par se emiten simultáneamente, mediante una modulación QPSK por ejemplo. Si las secuencias se entrelazan o concatenan para emitirlas con una modulación BPSK, aparecen interferencias en las IFW, aunque de menor magnitud que las obtenidas en la IW.

3.5.2. Influencia de la recepción parcial de los códigos T-ZCZ en la cota de auto-correlación

En el caso de pares T-ZCZ, cuando el porcentaje perdido de las secuencias del par supera el 50 % y $M > 4$ se obtienen valores de cota θ_{AC} superiores a 0.5. Estos valores de θ_{AC} se deben al aumento de los picos laterales de auto-correlación en la IW, pero también a la aparición de interferencias en las IFW. Obsérvese la diferencia con los códigos LS, en donde a pesar de aparecer interferencias alrededor del origen, éstas son de valor reducido en comparación con las obtenidas en la IW, incluso cuando el porcentaje de secuencia perdido es elevado. Sin embargo, cuando se pierde la mitad de un código T-ZCZ con $M > 4$ aparecen interferencias de valor similar a la mitad del pico principal en una zona alrededor del origen de tamaño igual al de la IFW, y también fuera de esa zona. Cuando $M \leq 4$, lo que implica un número de usuarios simultáneos $\mu \leq 4$, la cota θ_{AC} a lo largo de toda la SACF es reducida: $\theta_{AC} \leq 0.5$ para secuencias de $L \geq 32$ y porcentajes de pérdida de bits del 80 %.

La figura 3.26 muestra las cotas de auto-correlación en función del porcentaje perdido para $\mu = 4$ usuarios simultáneos y códigos de longitud a) $L = 32$ y b) $L = 512$ bits obtenidos a partir del método 2. Cuando se reciben los códigos completos, se obtienen lóbulos principales de valor 64 y 1024 respectivamente. En línea discontinua se ha indicado la cota θ_{AC} obtenida en lo que debería ser la IFW alrededor del origen. En la figura 3.27 puede observarse la cota θ_{AC} para $\mu = 8$ usuarios simultáneos y códigos de longitud $L = 512$.



(a) $L = 32$ (b) $L = 512$

Figura 3.26: θ_{AC} en función del porcentaje de secuencia perdido para $\mu = 4$ códigos T-ZCZ.

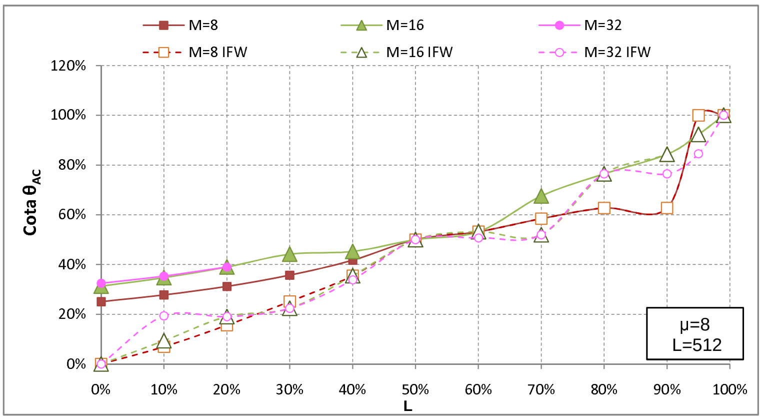


Figura 3.27: θ_{AC} para $\mu = 8$ códigos T-ZCZ simultáneos en función del porcentaje de secuencia perdido.

3.6. Conclusiones

En este capítulo se han analizado distintos esquemas de codificación con secuencias binarias, para su aplicación en un sistema de compresión de pulsos en el que puedan tener lugar múltiples emisiones simultáneas, aperiódicas y sin necesidades de un sincronismo estricto. Bajo estas premisas, las familias de códigos evaluadas han sido: Kasami, macro-secuencias obtenidas a partir del entrelazado o concatenación de CSS, LS y pares T-ZCZ.

Para todos ellas se ha realizado un proceso de búsqueda exhaustiva con el fin de encontrar la combinación de emisores que da lugar a las menores cotas de correlación θ, mostrando en cada caso un ejemplo de dicha combinación. Siguiendo la terminología acuñada en [SP80] para secuencias-m, se añadirá en lo sucesivo el apellido *preferidos* a la combinación de códigos de cualquier familia que reduzca al máximo el valor de la cota θ. Además, se ha estudiado la capacidad de cada familia de reducir la zona ciega que aparece cuando se utiliza un único transductor como emisor y como receptor.

En un sistema completamente asíncrono, los códigos Kasami son los más adecuados debido a sus valores bajos de cota de correlación, incluso cuando aumenta el número de usuarios simultáneos. Es más, de los códigos evaluados, son los que mejor se ajustan a la cota de Levenshtein. Además, puede reducirse el tamaño de la zona ciega incluso con secuencias largas: con códigos de longitud $L \leq 255$ la cota de auto-correlación es inferior a $\theta_{AC} = 0.5$ recibiendo sólo en torno al 20 % de la señal.

En cuanto a los CSS, se ha comprobado que las propiedades ideales de correlación de estos conjuntos se degradan cuando se utilizan mecanismos de transmisión adaptados a transductores acústicos de bajo ancho de banda. Específicamente, los mecanismos evaluados consisten en formar macro-secuencias concatenando o entrelazando los bits de los CSS y emitirlas posteriormente mediante una modulación BPSK. Para reducir al máximo el efecto de la ordenación de bits en las propiedades de complementaridad de los CSS se ha hecho una selección de aquellas de menor cota θ, comprobando que en la mayoría de los casos las μ macro-secuencias con menor interferencia mutua no han sido obtenidas a partir de CSS incorrelados. Para ambos métodos, los peores valores de cota θ se obtienen cuando en los CSS originales el número de secuencias coincide con su longitud ($M = L$); además, un aumento en el número de usuarios se traduce en un aumento de la cota θ. Las macro-secuencias generadas mediante concatenación tienen valores de interferencias más bajos que las obtenidas mediante entrelazado y reducen en mayor medida la zona ciega. En la figura 3.28 pueden compararse las cotas θ obtenidas con macro-secuencias y códigos Kasami. A pesar de que las macro-secuencias presentan valores más altos de cota, cuando la longitud de dichas macro-secuencias es elevada y el número de usuarios simultáneos es relativamente bajo en comparación con la longitud de los CSS iniciales, se justifica su empleo debido a la posibilidad de utilizar los correladores eficientes de CSS de los cuales derivan. Por otro lado, las macro-secuencias permiten una mayor variedad de longitudes y familias pseudo-incorreladas con un mayor número de códigos.

En aplicaciones cuasi-síncronas, en las que se puede garantizar que la diferencia de tiempos de llegada entre los ecos no excede un determinado límite, interesa el uso de códigos ortogonales generalizados ya que son capaces de eliminar completamente las ISI y MAI dentro de ese límite. Así, se han evaluado dos tipos de códigos: LS y pares T-ZCZ. Se ha comprobado que los códigos LS generados a partir de CSS presentan valores de cota de correlación menores en el área de interferencias que los generados a partir de Golay, aunque para obtener zonas libres de interferencias del mismo tamaño hay que recurrir a códigos más

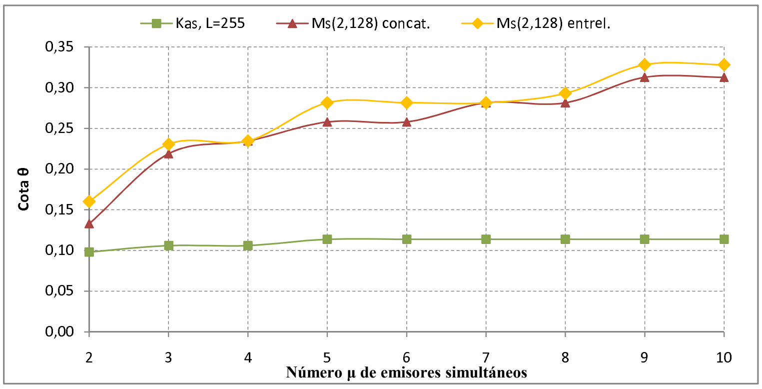


Figura 3.28: Comparativa de la cota θ obtenida con secuencias Kasami y macro-secuencias de 2-CSS, en función del número de usuarios simultáneos.

largos. Comparando estos códigos con parejas T-ZCZ con un pico principal de SACF del mismo valor, se tiene que la zona libre de interferencias de los LS es mayor, y que además estos últimos reducen en mayor medida el tamaño de la zona ciega. Por contrapartida, cuando el número de usuarios simultáneos está próximo al máximo que permite la familia, se obtienen valores de cota θ menores en la IW con los pares T-ZCZ, como puede observarse en la figura 3.29.

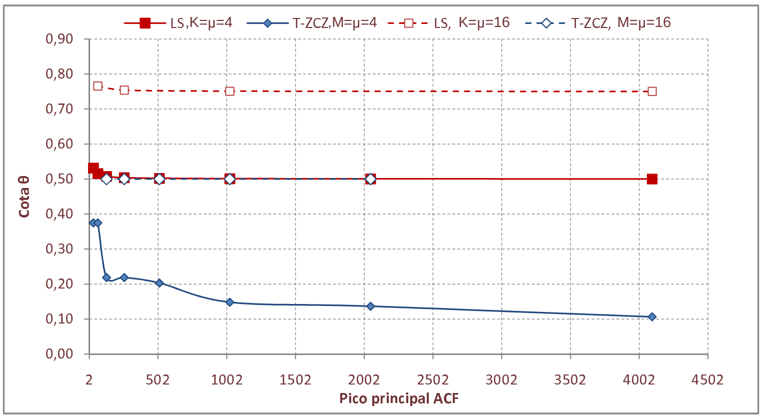


Figura 3.29: Comparativa de la cota θ obtenida en la IW frente al pico principal de auto-correlación con para códigos LS y T-ZCZ.

Capítulo 4

Implementación eficiente de correladores para secuencias CSS y LS

Ya en el capítulo 2 se mencionaba que, para detectar una señal contaminada con ruido blanco optimizando la relación señal-ruido a la salida, puede emplearse un filtro acoplado cuya respuesta impulsiva sea proporcional a una versión reflejada de la señal que se desea detectar. El filtro realiza la correlación de la señal emitida ideal con la que llega afectada por el ruido, de modo que emitiendo códigos con propiedades adecuadas de correlación, se pueden localizar fácilmente dichos códigos en la señal recibida.

Una implementación directa y sencilla de este filtro consiste en un correlador paralelo que en un único ciclo de reloj sume el resultado de multiplicar las L últimas muestras recibidas con las L muestras de la secuencia patrón. Ahora bien, en aplicaciones que requieren longitudes de secuencia elevadas resulta inviable el procesamiento en tiempo real debido al excesivo número de operaciones a realizar. Por otro lado, un correlador serie que realice una única operación por ciclo de reloj es demasiado lento. Una alternativa para reducir el número de operaciones consiste en utilizar técnicas de correlación basadas en el algoritmo FFT [JB98]; pero implican la realización de multiplicaciones, a diferencia de las implementaciones anteriores en donde estas operaciones pueden reducirse a sumas y restas. Como las multiplicaciones son más costosas de realizar que las sumas y restas, una solución basada en FFTs tampoco es la más óptima.

La búsqueda de soluciones a estos inconvenientes ha potenciado la investigación de algoritmos de detección eficientes asociados a códigos concretos que permitan reducir el número de operaciones a realizar, normalmente a costa de utilizar un mayor número de elementos de almacenamiento.

En cuanto a correladores asociados a secuencias pseudo-aleatorias destacan los propuestos en [Bud89] y [Pop97] para secuencias-m y Gold ortogonales respectivamente. La ventaja de estos correladores frente a los basados en la FFT es que aprovechan la relación existente entre las secuencias-m y las Walsh para reducir, gracias a la FWT, el número de operaciones sin que ello implique la resolución de multiplicaciones. Por contra, este algoritmo FWT realiza la correlación periódica y es de poco interés en sistemas que usan un mismo transductor como emisor y receptor.

El correlador eficiente de códigos Golay (EGC) [Bud91, Pop99] es una buena alternativa a los anteriores: puede emplearse para detección aperiódica, realiza un menor número de operaciones que un correlador directo[1] y éstas se reducen a sumas y restas cuando se trabaja con códigos binarios. Un trabajo posterior, [DMUH$^+$07b], propone un esquema más general que permite la correlación de cualquier número $M = 2^m$ de secuencias complementarias de longitud $L = M^N$ (recuérdese que las parejas Golay pueden considerarse un caso especial de CSS con dos secuencias). El problema de este algoritmo reside en su implementación práctica, resultando complicado realizar un diseño genérico debido a la estructura de sus conexiones internas. En este capítulo se va a proponer una modificación a dicho algoritmo, para facilitar su implementación hardware genérica con independencia del número de secuencias del conjunto a detectar y de su longitud. Además, el correlador se adaptará al esquema de emisión mediante macro-secuencias, y se llevará a cabo un estudio para determinar qué mecanismo de ordenación de bits permite una correlación más rápida y con menos recursos. De este modo, aunando los resultados del capítulo anterior con los de éste, puede seleccionarse el grupo de macro-secuencias de CSS con mejores prestaciones tanto desde el punto de vista de sus propiedades de correlación, como de la implementación hardware de su detector asociado.

Se propone además en este capítulo un novedoso algoritmo de correlación para códigos LS basado en la relación de estos códigos con los CSS. Por otro lado, en la búsqueda de métodos eficientes de correlación de códigos T-ZCZ se ha llegado a una nueva propuesta de codificación con propiedades de correlación similares, e incluso mejores para determinadas longitudes que las presentadas por los códigos T-ZCZ descritos en [ZLH04, ZLH05], y con la posibilidad de reducir en un alto porcentaje el número de operaciones en la generación y detección. Para mayor claridad, este nuevo esquema de codificación y el correlador eficiente desarrollado se explican en el siguiente capítulo.

En varios trabajos previos [GN98, Her03, Á05] se ha mostrado que, para procesar en tiempo real señales ultrasónicas mediante técnicas de compresión de pulsos similares a las propuestas en esta tesis, la implementación más adecuada es la basada en una arquitectura

[1] En un correlador directo convencional, ya sea serie o paralelo [PG93], el coste computacional es proporcional a L y resulta de evaluar directamente las ecuaciones: $R_{a_m,a_s}[\tau] = \sum_{l=0}^{L-1} a_m[l]a_s[l+\tau]$ en el caso de emisión periódica ó $C_{a_m,a_s}[\tau] = \sum_{l=0}^{L-1-\tau} a_m[l]a_s[l+\tau]$ si la emisión es aperiódica. Nótese que las operaciones de multiplicación se simplifican en sumas y restas dado el carácter binario de los códigos empleados.

configurable. Por este motivo, en este capítulo se muestran resultados de implementación de cada uno de los algoritmos propuestos sobre sistemas de este tipo y, más concretamente, sin pérdida de generalidad, sobre FPGAs de Xilinx [Xil08d].

4.1. Opciones de implementación de un correlador directo

El objetivo de esta sección es el de obtener una implementación básica de un correlador directo convencional, para poder usarla como referencia de comparación con los algoritmos eficientes desarrollados.

Un correlador paralelo implementado únicamente con lógica combinacional se descarta, ya que sólo resulta viable para secuencias cortas ($L \leq 64$) y ancho de bus de datos de entrada reducido. Otra posibilidad es un diseño serie; en este sentido se han evaluado tres opciones distintas, cuya variación reside en el modo de gestionar la memoria necesaria para almacenar las muestras de entrada en la FPGA.

Una primera solución haría uso de los biestables de los *slices* de la FPGA. Los datos de entrada se reciben con una frecuencia f_L y almacenan en un buffer de tamaño igual a la longitud L del código con el que se desea realizar la correlación (en adelante código patrón). La frecuencia de lectura del buffer es $f_H = L \cdot f_L$, de modo que por cada muestra de entrada se accede L veces al buffer. En cada acceso, la muestra leída se suma o resta al resultado acumulado anterior en función del código patrón. Cuando llega una nueva muestra al buffer, las muestras anteriores se desplazan una posición perdiéndose la más antigua, y el valor acumulado se pone a cero para efectuar una nueva correlación.

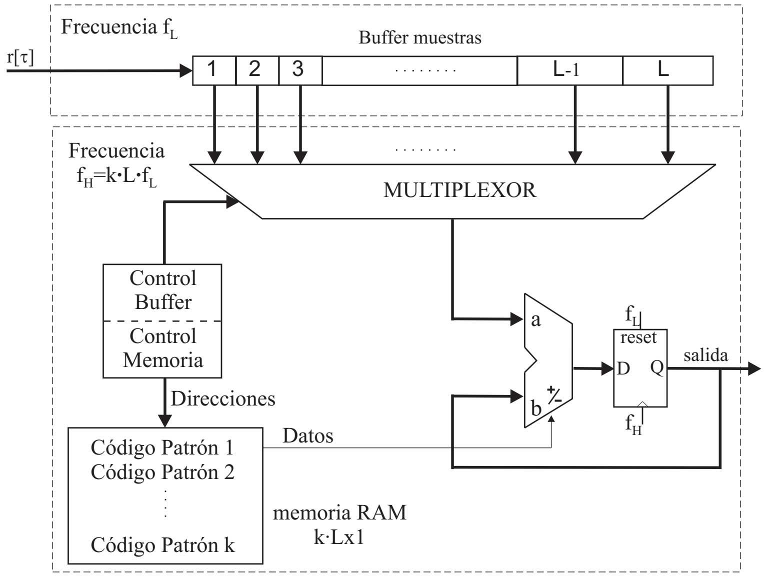
Si en el sistema hay k emisores simultáneos, cada uno codificado mediante un código patrón distinto, se necesitan k correladores para discriminar en la secuencia recibida cada uno de estos códigos. En estos casos, las correlaciones pueden hacerse de modo secuencial, adaptando el sistema descrito anteriormente para leer k veces el buffer y llevar a cabo en cada lectura la correlación con el código patrón que corresponda. Se reducen así las necesidades de hardware, pero aumenta la frecuencia f_H del bloque computacional, de modo que $f_H = k \cdot L \cdot f_L$ (véase la figura 4.1.a). Otra posibilidad es realizar las correlaciones en paralelo: el bloque computacional encargado de realizar las sumas y restas de la correlación se replica k veces, mientras que el buffer de muestras es compartido por todos ellos, como puede observarse en la figura 4.1.b. En este caso los requisitos de frecuencia son menos exigentes ($f_H = L \cdot f_L$) y, aunque aumenta el número de recursos, no se trata de un aumento importante ($k-1$ sumadores/restadores y $k-1$ biestables tipo D adicionales). Sin embargo, sí que existe un problema de consumo de recursos, pero es común a ambos diseños y se debe a la implementación del buffer de almacenamiento. Este buffer hace uso de los biestables contenidos en los *slices* de la FPGA, en lugar de configurar los módulos generadores de funciones de dichos *slices* como registros de desplazamiento serie de 16 bits (SRL16) o memoria SelectRAM (RAM16) también de 16 bits. Esto conlleva tener una capacidad de almacenamiento de 2 bits por *slice* en lugar de 32 bits[2], lo que redunda en un consumo excesivo de los ya mencionados *slices* que puede dar lugar a la necesidad de trabajar con FPGAs de mayor tamaño y por ende de mayor coste.

Por otro lado, considerando un sistema práctico ultrasónico de compresión de pulsos, el código a emitir debe ser modulado para adaptarlo al rango de frecuencias del transductor. Si no se dispone de ninguna referencia temporal, la demodulación se realiza de forma asíncrona muestreando la señal recibida con una frecuencia lo suficientemente elevada y correlándola digitalmente con el símbolo empleado en la modulación. Si O_f es el factor de sobremuestreo y N_{SM} el número de períodos del símbolo de modulación, los bits correspondientes al código emitido se obtienen con una separación de $O_f \cdot N_{SM}$. Sabiendo esto, el nuevo buffer de almacenamiento de muestras del correlador debe tener una capacidad de $L \cdot O_f \cdot N_{SM}$ posiciones de tamaño DW; en donde DW es el ancho de bus de datos resultante tras el proceso de conversión analógico-digital y la demodulación. El acceso a dicho buffer debe realizarse en saltos de $O_f \cdot N_{SM}$ posiciones. En estos casos, la implementación del buffer en la memoria interna distribuida de la FPGA y, más concretamente en los biestables de los *slices*, no es una buena solución ya que implica un elevado consumo de recursos.

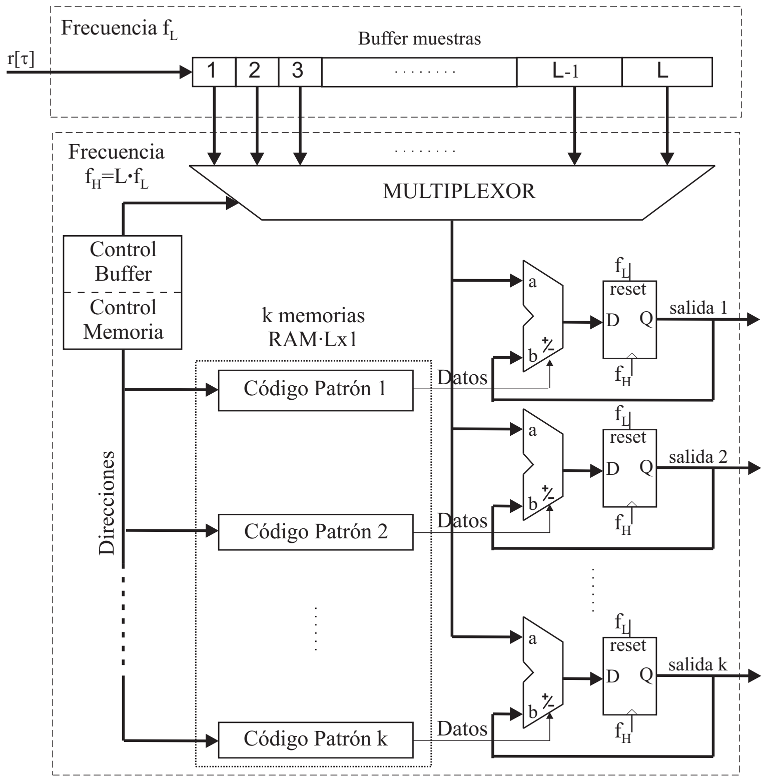
Como solución, el buffer de muestras puede implementarse en una memoria externa a la FPGA o en los bloques de memoria SelectRAM dedicados de la FPGA. En ambos casos se reduce considerablemente el número de *slices* consumidos, a costa de complicar en cierta medida el gestor de memoria que controla los accesos al buffer. A modo de ejemplo la tabla 4.1 muestra los recursos y frecuencias máximas de operación para la detección de un único código Kasami de longitud $L = 255$ bits usando una arquitectura como la de la figura 4.1 y una FPGA Virtex-II Pro de Xilinx [Xil07]. No se han considerado parámetros dependientes de la modulación empleada o del conversor utilizado, esto es, $N_{SM} = O_f = DW = 1$. En la tabla, la primera columna de resultados corresponde al caso de implementar el buffer de muestras en los biestables de los *slices* de la FPGA; la siguiente columna al caso de emplear una memoria RAM externa a la FPGA; y en la última se muestran las necesidades hardware cuando se usan bloques de memoria SelectRAM RAMB16 de la FPGA configurados como memoria de un único puerto. En los tres casos se ha especificado la frecuencia máxima de procesamiento de las muestras del buffer (f_H) y la frecuencia máxima de obtención de cada nuevo resultado de correlación (f_L). Puede observarse que el diseño más óptimo es el que utiliza los bloques de memoria RAM internos de la FPGA: por un lado reduce el consumo de *slices*; y por otro, al no tener que efectuar accesos a memoria externa simplifica el gestor de memoria y éstos pueden hacerse a mayor velocidad.

Evidentemente los diseños propuestos aquí no son la única solución para implementación

[2]Véase [ML91] para detalles acerca de la estructura genérica de una FPGA y [Xil08a] para el caso de una familia característica de bajo coste como es la Spartan-3 de Xilinx. FPGAs de Xilinx de última generación, como las Virtex-5 [Xil08c], tienen una capacidad mayor de almacenamiento por *slice*, pero continúa existiendo una diferencia importante entre usar los generadores de funciones como memoria RAM distribuida o registros de desplazamiento serie, frente a usar los elementos de almacenamiento dedicados.



(a) Implementación secuencial de las k correlaciones.



(b) Implementación paralelo de las k correlaciones.

Figura 4.1: Implementación de un correlador directo serie con k códigos patrón.

xc2vp100	Memoria slices	Memoria RAM externa	Memoria RAM interna
Slices	303	68	68
LUTs	329	112	110
IOBs	13	24	13
Bloques RAM	1	1	2
f_H	104.92 MHz	130.907 MHz	147.929 MHz
f_L	411.45 KHz	513.36 kHz	580.113 kHz
Tiempo de correlación ($\frac{1}{f_L}$)	2.43 μs	1.95 μs	1.72 μs

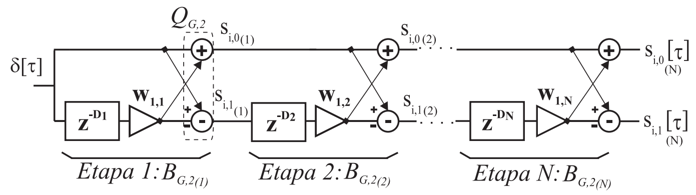
Tabla 4.1: Comparativa de recursos y tiempos de ejecución empleando un correlador directo serie para la detección de un código Kasami de $L = 255$ bits.

de un correlador directo. Podría pensarse por ejemplo en almacenar las muestras a procesar en bloques de memoria internos de la FPGA configurados como memorias de doble puerto, de modo que por un puerto se accediese a las muestras más antiguas para realizar su procesamiento y por el otro a las más recientes, sumando posteriormente el resultado de ambas correlaciones parciales. Con esto se conseguiría duplicar la frecuencia de correlación, pero serían necesarios más recursos hardware. Por su simplicidad y reducción de los recursos necesarios para realizar la correlación se ha optado por la implementación descrita anteriormente basada en bloques SelectRAM internos a la FPGA, para su posterior comparación con los diseños eficientes propuestos en esta tesis.

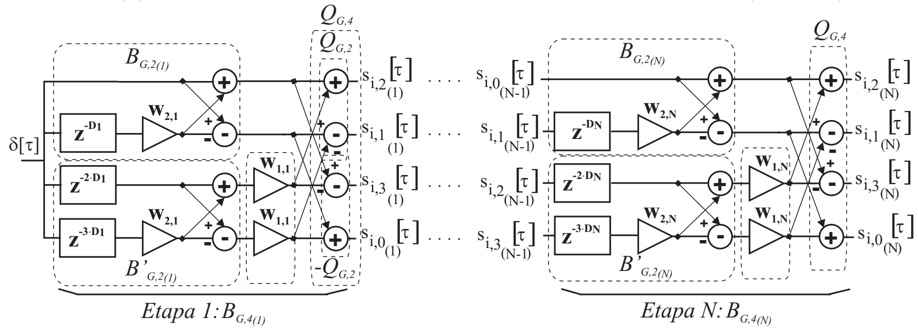
4.2. Propuesta de implementación de un correlador eficiente de CSS

En [DMUH+07b] se muestra un método optimizado para la obtención y procesamiento de CSS. Este método constituye una generalización de los propuestos para parejas Golay (2-CSS) [Bud91, Pop99] y conjuntos de cuatro secuencias complementarias (4-CSS) [ÁUM+04], permitiendo la construcción de generadores y correladores eficientes de conjuntos de cualquier número $M = 2^m$ de secuencias complementarias de longitud $L = M^N$, donde $m, N \in \mathbb{N} - \{0\}$. El apéndice B resume el mecanismo de generación de las ecuaciones recursivas que permiten la obtención de CSS según dicho trabajo; y en la figura 4.2 se muestra el diagrama de bloques de generadores eficientes de conjuntos de (a) dos secuencias complementarias (2-ESSG, *Efficient Set of 2 Sequences Generator*), (b) cuatro secuencias complementarias (4-ESSG) y (c) M secuencias complementarias (M-ESSG).

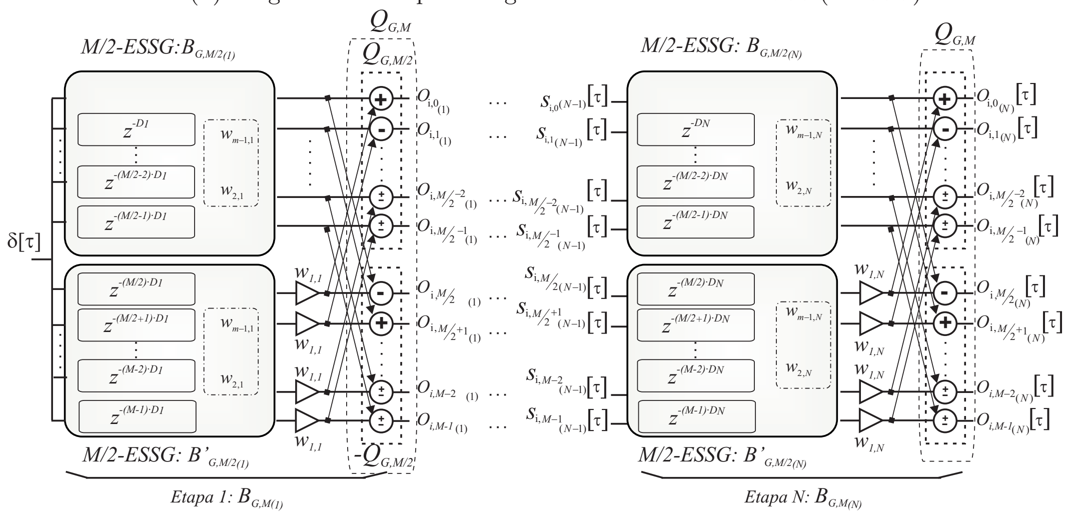
Según se observa en la figura 4.2, el generador puede considerarse como un filtro digital



(a) Diagrama de bloques del generador eficiente de 2-CSS (2-ESSG).



(b) Diagrama de bloques del generador eficiente de 4-CSS (4-ESSG).



(c) Diagrama de bloques del generador eficiente de M-CSS (M-ESSG) obtenido a partir de un generador de M/2-CSS (M/2-ESSG).

Figura 4.2: Diagrama de bloques del generador eficiente de conjuntos de secuencias complementarias propuesto en [DMUH+07b], para a) 2-CSS, b) 4-CSS y c) M-CSS.

de N etapas similares, cada una de las cuales posee una serie de elementos de retardo, sumadores, restadores y multiplicadores. Para que las secuencias generadas sean binarias, con valores 1 y -1, los retardos D_n deben escogerse como cualquier permutación del conjunto $\{M^0, M^1, M^2, \cdots, M^{N-1}\}$. Es más, si se desea evitar el desbordamiento tras las operaciones internas de cada etapa, son necesarios m bits adicionales de memoria en cada etapa para cada muestra almacenada. Entonces, eligiendo para los retardos D_n la permutación inversa $\{M^{N-1}, M^{N-2}, \cdots, M^0\}$, de modo que a la primera etapa le correspondan los retardos de mayor valor, puede reducirse la cantidad de memoria total necesaria para la implementación del generador.

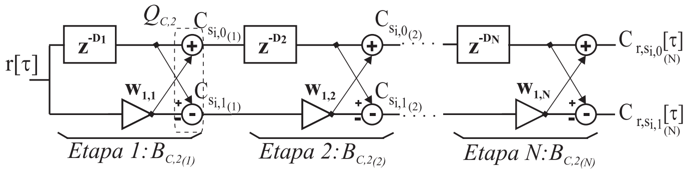
Los coeficientes $W_N^{(m} = (w_{1,1}, w_{2,1}, \cdots, w_{m,1}, w_{1,2}, w_{2,2}, \cdots, w_{m,2}, \cdots, w_{1,N}, \cdots, w_{m,N})$ constituyen la semilla de generación del conjunto y pueden tomar valores 1 ó -1. Quedan reducidas por tanto a sumas y restas las operaciones de multiplicación por estos coeficientes. Además, con este esquema de generación es muy sencilla la obtención de M CSS mutuamente incorrelados: basta con realizar todas las posibles combinaciones de los coeficientes $(w_{1,1}, w_{2,1}, \cdots, w_{m,1})$ de la primera etapa y fijar el resto de coeficientes a un valor determinado. Por otro lado, y para mayor comodidad, el vector con los coeficientes $w_{k,n}$ se puede representar en base decimal como $Wd_N^{(m}$ considerando los valores -1 como 0 y siendo $w_{1,1}$ el bit más significativo.

Obsérvese también que el filtro posee M salidas por donde se obtienen las secuencias complementarias cuando la entrada al sistema es la secuencia impulsiva $\delta[\tau]$. Sabiendo que la respuesta impulsiva de un filtro a una secuencia determinada es la versión reflejada en el tiempo de esa secuencia, puede verse el sistema anterior como un filtro acoplado a las versiones reflejadas en el tiempo de las secuencias complementarias $s_{i,j}[L-1-\tau]$, donde $L = M^N$ es la longitud de las secuencias. Permutando el orden de los retardos D_n en cada etapa del esquema de generación se obtienen las secuencias reflejadas, y el nuevo filtro equivale a un correlador acoplado a las secuencias directas. Este filtro, denominado correlador eficiente de CSS (ESSC, *Efficient Set of Sequences Correlator*), posee M salidas que corresponden a la correlación simultánea de la señal de entrada con las M secuencias complementarias de un conjunto S_i, según puede observarse en la figura 4.3 para (a) 2-CSS, (b) 4-CSS y (c) M-CSS.

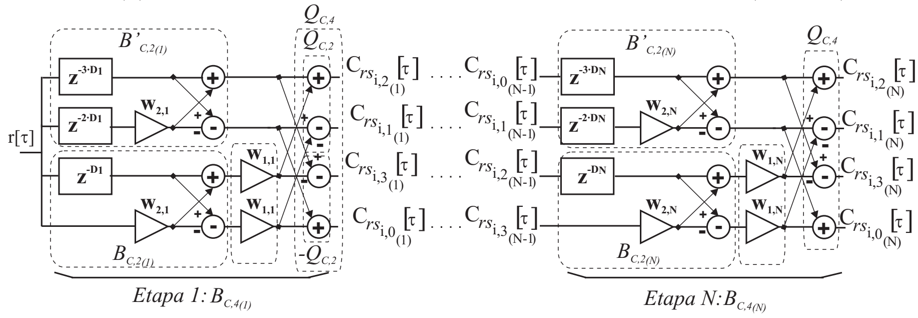
La ventaja de este correlador ESSC frente a una implementación directa tradicional reside en la reducción en el número total de operaciones efectuadas. En la versión clásica son necesarias $M \cdot L$ multiplicaciones y $M \cdot (L-1)$ sumas para la correlación de la señal de entrada con las M secuencias del conjunto; mientras que utilizando el ESSC sólo se realizan $\frac{M}{2} \cdot log_2 L$ multiplicaciones y $M \cdot log_2 L$ sumas. La profundidad de memoria necesaria para almacenar los datos es mayor en el ESSC, aunque el número de accesos a memoria por cada resultado de correlación se reduce desde 2^{mN} a $(2^m - 1)N$ (considerando que los accesos de lectura y escritura pueden solaparse). La tabla 4.2 recoge estas ideas, teniendo en cuenta que DW es el número de bits necesarios para representar los datos de entrada, O_f el factor de sobremuestreo, N_{SM} el número de períodos del símbolo de modulación y que se ha usado para la comparación un correlador directo serie como el de la figura 4.1.b. Conviene recordar que, debido al uso de secuencias binarias con valores en el conjunto $\{-1, 1\}$, las multiplicaciones pueden simplificarse en sumas y restas.

Tanto en el caso del generador ESSG como del correlador ESSC, la obtención de la arquitectura para M secuencias complementarias hace uso de dos arquitecturas de $M/2$, teniendo en cuenta las siguientes consideraciones:

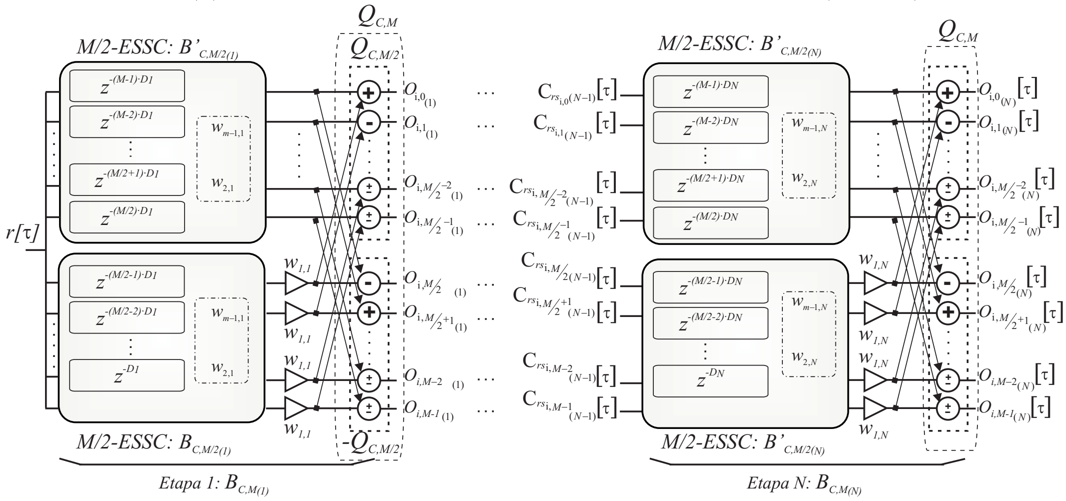
- En el caso del generador M-ESSG, los coeficientes $\{0, 1, 2, \cdots, M/2-1\}$ que multiplican los retardos D_n del bloque superior $B_{G,M/2}$ permanecen como están en el



(a) Diagrama de bloques del correlador eficiente de 2-CSS (2-ESSC).



(b) Diagrama de bloques del correlador eficiente de 4-CSS (4-ESSC).



(c) Diagrama de bloques del correlador eficiente de M-CSS (M-ESSC) obtenido a partir de un correlador de M/2-CSS (M/2-ESSC).

Figura 4.3: Diagrama de bloques del correlador eficiente de conjuntos de secuencias complementarias propuesto en [DMUH+07b], para a) 2-CSS, b) 4-CSS y c) M-CSS.

Implementación	Productos	Sumas	Bits de Memoria	Accesos a memoria
Correlador directo serie	$M \cdot L$	$M \cdot (L-1)$	$N_{SM} \cdot O_f \cdot DW \cdot M^N + M^{N+1} + M \cdot (DW + mN)$	2^{mN}
M-ESSC	$\frac{M}{2} \cdot log_2 L$	$M \cdot log_2 L$	$N_{SM} \cdot O_f \cdot DW \cdot (M-1)M^{N-1} + N_{SM} \cdot O_f \frac{M^2-M}{2} \left[DW \frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right]$	$(2^m - 1)N$

Tabla 4.2: Comparativa de las necesidades computacionales de un correlador directo serie y el modelo ESSC para la correlación de la señal de entrada con las M secuencias de un CSS.

generador de M/2-ESSG; pero los retardos del bloque inferior $B'_{G,M/2}$ toman valores del conjunto $\{M/2, M/2+1, \cdots, M-1\}$, siendo $(M-1)$ el retardo correspondiente a la rama inferior del generador. Como resultado, los coeficientes de cada etapa B_G del M-ESSG son, ordenados según aparecen desde la rama superior a la inferior, $\{0, 1, \cdots, M-1\}$.

En el correlador sucede algo similar, sólo que ahora el bloque que debe modificar el valor de sus retardos por $\{(M-1), (M-2), \cdots, (M/2+1), M/2\}$ es el superior $B'_{C,M/2}$; mientras que los coeficientes que acompañan a los retardos D_n del bloque inferior $B_{C,M/2}$ son los mismos que en la estructura del M/2-ESSC ($\{(M/2-1), \cdots, 2, 1, 0\}$). Con esto, los coeficientes de cada etapa B_C del M-ESSC son $\{M-1, M-2, \cdots, 1, 0\}$.

- Pasar de la arquitectura de $M/2$ a una de M implica añadir un nuevo bit de semilla $w_{1,n}$ en cada etapa $1 \leq n \leq N$ del generador y del correlador. Este nuevo bit se traduce en $M/2$ multiplicadores en cada salida del bloque inferior $B'_{G,M/2}$ o $B_{C,M/2}$, según se trate del generador o del correlador.

- Resulta necesario agregar además una nueva etapa de sumadores/restadores. En el generador, los $M/2$ sumadores/restadores del bloque superior tienen el mismo signo que los sumadores/restadores $Q_{G,M/2}$ del $M/2-ESSG$. Sin embargo, los $M/2$ sumadores/restadores del bloque inferior tienen el signo invertido, esto es: $-Q_{G,M/2}$. Lo mismo sucede en el caso del correlador.

- Por último, se deben reordenar las salidas de cada etapa antes de conectarlas a la siguiente ya que no existe una correspondencia directa. Así, la relación T_m existente entre la posición de la salida x de una etapa cualquiera (considerando que la salida 0 es la de la rama superior) y la secuencia j se obtiene según (4.1):

$$\begin{aligned}
T_{m=1} &= \{(x,j)/x \in \{0,1\} \wedge j \in \{0,1\} \wedge x = j\} = \{(0,0),(1,1)\} \\
T_{m\geq 2} &= \left\{ (x,j) \middle/ \begin{array}{l} (\exists \xi \in \{1,2,\cdots,2^{m-2}\}) \\ (x = 4\xi - 1 \wedge (2\xi-2, j) \in T_{m-1}) \vee (x = 4\xi - 2 \wedge j = 2^m - z \wedge (2\xi-1, z) \in T_{m-1}) \vee \\ (x = 4\xi - 3 \wedge (2\xi-1, j) \in T_{m-1}) \vee (x = 4\xi - 4 \wedge j = 2^m - z \wedge (2\xi, z) \in T_{m-1}) \end{array} \right\}
\end{aligned} \tag{4.1}$$

Donde $M = 2^m$ indica el número de secuencias del conjunto y $m \in \mathbb{N} - \{0\}$. Conociendo la relación entre el número x de salida en una etapa y la secuencia $s_{i,j}$ correspondiente, puede conectarse la salida obtenida en la etapa $n-1$ con la entrada j de la etapa n. Para mayor claridad la tabla 4.3 muestra el orden de salida en una etapa cualquiera para el caso de 2-CSS, 4-CSS, 8-CSS y 16-CSS.

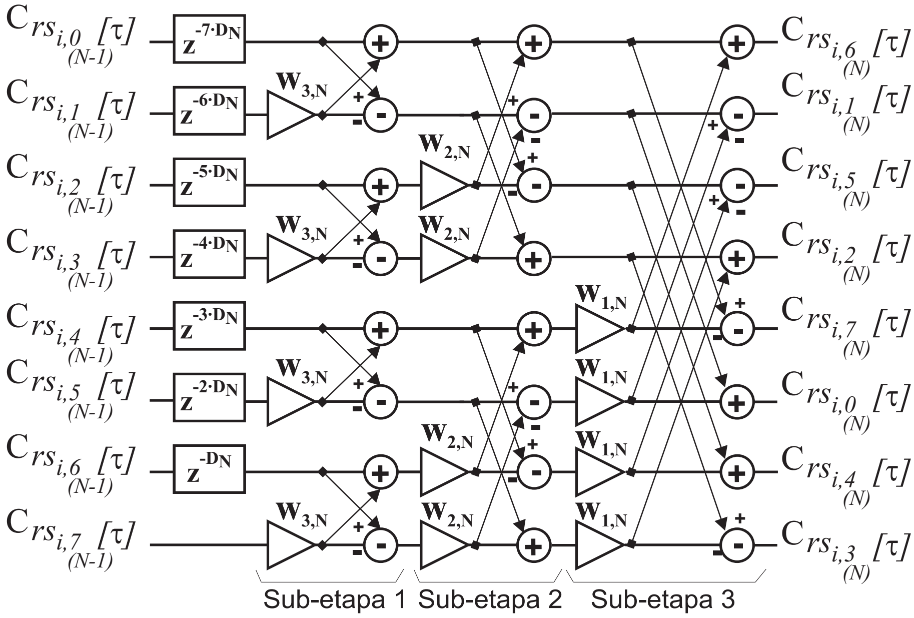
Las arquitecturas descritas para el M-ESSG y el M-ESSC no son las más adecuadas para una implementación genérica en hardware, ya que las conexiones internas de cada etapa y el orden en el que aparecen las secuencias a la salida de dichas etapas dependen del número M de secuencias del conjunto. Específicamente, implementar en hardware la ecuación (4.1)

Nº de salida de la etapa (x)	Índice j de la secuencia $s_{i,j}$ complementaria correspondiente			
	$M = 2$	$M = 4$	$M = 8$	$M = 16$
0	0	2	6	14
1	1	1	1	1
2	–	3	5	9
3	–	0	2	6
4	–	–	7	13
5	–	–	0	2
6	–	–	4	10
7	–	–	3	5
8	–	–	–	15
9	–	–	–	0
10	–	–	–	8
11	–	–	–	7
12	–	–	–	12
13	–	–	–	3
14	–	–	–	11
15	–	–	–	4

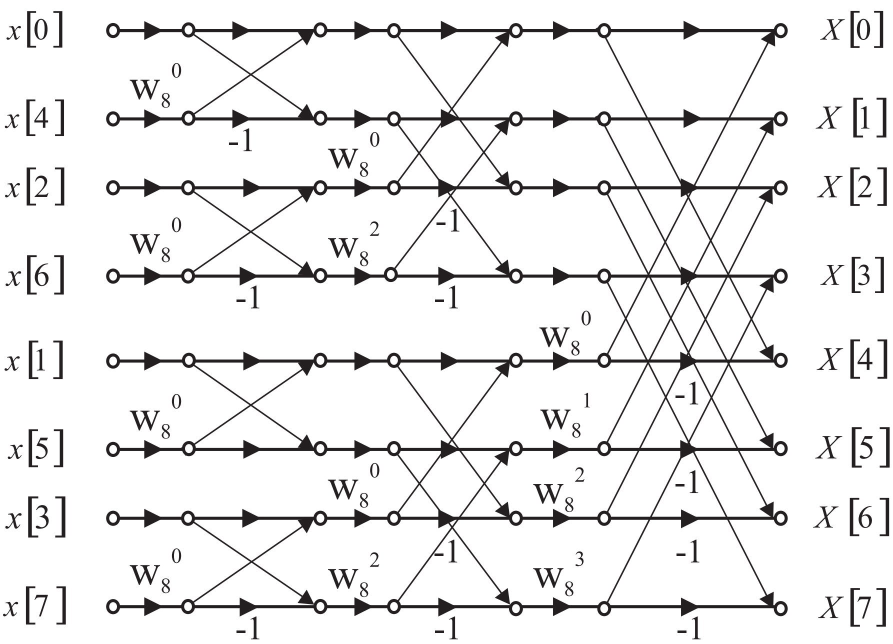
Tabla 4.3: Correspondencia entre las salidas de una etapa y la secuencia $s_{i,j}$ o la correlación parcial $C_{r,s_{i,j}}$ obtenida, según se trate del M-ESSG o del M-ESSC propuesto en [DMUH+07b].

de modo que sea válida para cualquier valor de $M = 2^m$ es una tarea ardua, que aumenta la complejidad del diseño y en consecuencia la cantidad de recursos y operaciones a realizar. Una solución más sencilla consistiría en almacenar en memoria el patrón de ordenación de las secuencias de salida para distintos valores de M, y utilizar en cada caso el que corresponda. Se propone aquí, sin embargo, una alternativa mejor, sin necesidad de recursos de memoria adicionales y sin pérdida de generalidad. Esta nueva propuesta modifica el orden de las entradas de cada etapa del generador y correlador descrito en [DMUH+07b] para obtener una arquitectura regular y con un sencillo patrón de ordenación de salidas, facilitando así una implementación hardware genérica capaz de adaptarse al número M de secuencias del conjunto y a su longitud L.

Se detallan a continuación los cambios a realizar en el correlador, que son extrapolables al caso del generador. Estos cambios se sirven de la similitud existente entre la estructura del M-ESSC (véase la figura 4.4.a) y la estructura de un algoritmo FFT de diezmado en el tiempo y cómputo en el mismo lugar [OS00] (véase la figura 4.4.b). Así, cambiando la disposición de las entradas y conexiones internas del correlador de modo similar a como se hace en la FFT se puede conseguir una configuración más adecuada.



(a) Diagrama de bloques de una etapa del 8-ESSC.



(b) Diagrama de flujo de una FFT de 8 puntos.

Figura 4.4: Semejanza entre una etapa del M-ESSC y el algoritmo FFT.

Se considera que el orden natural de aparición de los coeficientes que multiplican los retardos D_n es $\{M-1, M-2, \cdots, 1, 0\}$, en donde el coeficiente $M-1$ corresponde al retardo de la rama superior. Además, el valor de estos coeficientes puede expresarse en formato binario de m bits. En la reestructuración propuesta los coeficientes que multiplican los retardos se reordenan leyendo su representación binaria al revés. Por ejemplo, en un 8-ESSC, el retardo $3 \cdot D_n$ aparecerá ahora en la posición que ocupaba anteriormente el retardo $6 \cdot D_n$, y viceversa. Este nuevo patrón de ordenación, que responde al nombre de *algoritmo de reflexión binaria*, queda representado en la tabla 4.4 para el caso específico de un 8-ESSC; según esto, la nueva estructura del 8-ESSC sería la mostrada en la figura 4.5.

$b_2 b_1 b_0$ → Reflexión Binaria	→ $b_0 b_1 b_2$
(000)	(000)
(001)	(100)
(010)	(010)
(011)	(110)
(100)	(001)
(101)	(101)
(110)	(011)
(111)	(111)

Tabla 4.4: Algoritmo de reflexión binaria para $M = 8$.

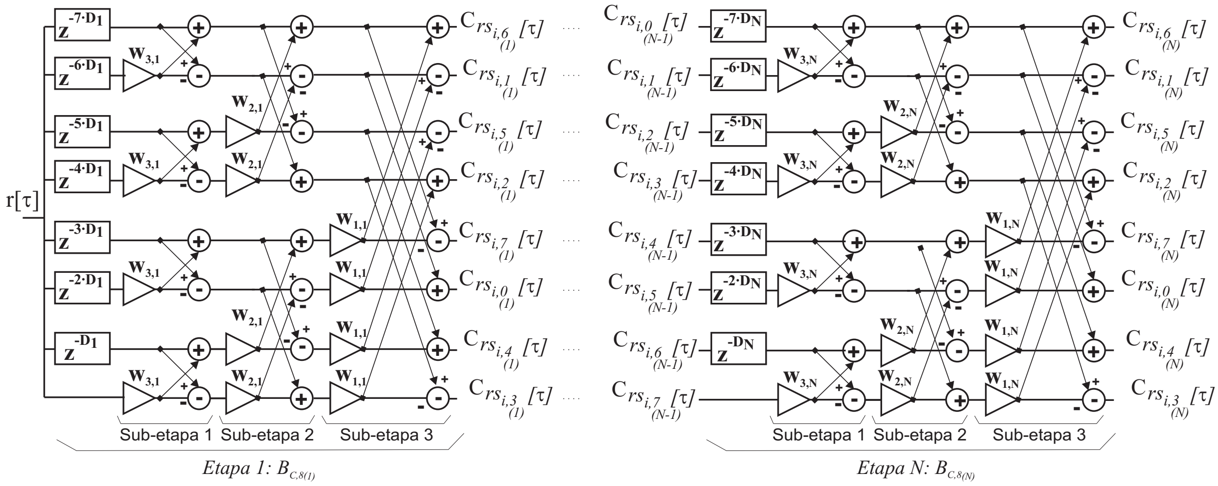
Obsérvese en la figura 4.5 que, tras la reordenación de retardos D_n, el resultado $C_{r,s_{i,j_{(n)}}}[\tau]$ de la correlación parcial de la señal de entrada $r[\tau]$ con la j-ésima secuencia $s_{i,j}[\tau]$ no se conecta a la j-ésima entrada, sino a la resultante de aplicar el algoritmo de reflexión binaria a j. Se consigue con esto que la estructura de las conexiones entre subetapas de una misma etapa sea la misma con independencia del número M de secuencias del conjunto. Además, las salidas de cada etapa se obtienen ahora siguiendo un patrón de ordenación fácilmente reproducible para cualquier valor de M. Considerando nuevamente que x indica la posición de una salida cualquiera en una etapa, siendo $x = 0$ la salida de la rama superior, y sabiendo que j es el índice que indica la correlación parcial $C_{r,s_{i,j_{(n)}}}[\tau]$ con la secuencia j-ésima, entonces la relación entre ambos viene dada por (4.2).

$$\begin{array}{ll} \text{Si } x \text{ par:} & j = (M - 2 - x) \\ \text{Si } x \text{ impar:} & j = (M - x) \end{array} \tag{4.2}$$

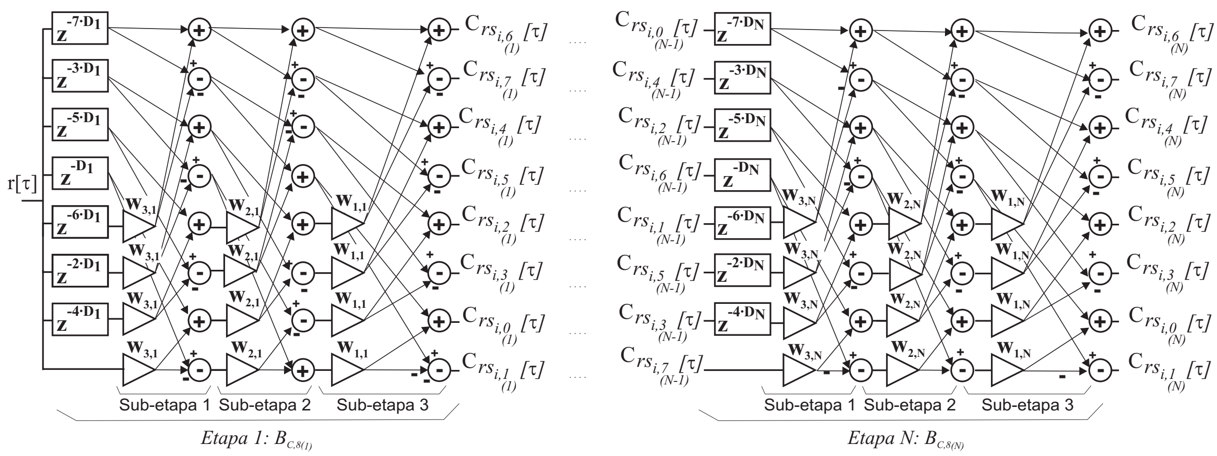
Reescribiendo la tabla 4.3 para el caso de aplicar el algoritmo de reflexión binaria se obtiene la tabla 4.5, calculada a partir de (4.2).

4.2.1. Opciones de configuración de los parámetros del M-ESSC

A la hora de implementar el correlador M-ESSC existen tres posibles estrategias de diseño a considerar:



(a) Diagrama de bloques del 8-ESSC propuesto en [DMUH+07b].



(b) Modificación de la figura 4.5.a para tener la misma geometría en cada sub-etapa y un sencillo patrón de ordenación de salidas.

Figura 4.5: Transformación de la estructura del M-ESSC propuesto en [DMUH+07b].

Nº de salida de la etapa (x)	Índice j de la secuencia $s_{i,j}$ complementaria correspondiente			
	$M = 2$	$M = 4$	$M = 8$	$M = 16$
0	0	2	6	14
1	1	3	7	15
2	–	0	4	12
3	–	1	5	13
4	–	–	2	10
5	–	–	3	11
6	–	–	0	8
7	–	–	1	9
8	–	–	–	6
9	–	–	–	7
10	–	–	–	4
11	–	–	–	5
12	–	–	–	2
13	–	–	–	3
14	–	–	–	0
15	–	–	–	1

Tabla 4.5: Correspondencia entre las salidas x de una etapa y la correlación parcial $C_{r,s_{i,j}}$ obtenida tras aplicar el algoritmo de reflexión binaria.

- Diseño a medida específico para un CSS determinado.
- Implementación configurable en pre-síntesis, en donde los parámetros del sistema se concretan al sintetizar el diseño.
- Implementación configurable en post-síntesis, en donde los parámetros del sistema son entradas del diseño.

La primera opción se descarta puesto que obliga a realizar un nuevo diseño cada vez que se desea cambiar algún parámetro del correlador. La segunda opción, aprovecha la reordenación de retardos propuesta, para realizar un diseño configurable capaz de realizar la correlación con conjuntos de cualquier número M de secuencias de longitud L. Ahora bien, los parámetros M y L deben ser concretados al sintetizar el diseño, de modo que una vez completadas las fases de síntesis e implementación se obtiene un diseño específico para la aplicación a realizar. La tercera opción permite sin embargo configurar los valores de M y L en tiempo de ejecución, a costa de utilizar un mayor número de recursos. Se detallan a continuación las bondades de estas dos últimas alternativas de diseño.

Configuración en pre-síntesis

Teniendo en cuenta las consideraciones anteriores, se ha realizado un primer diseño en torno a cinco parámetros genéricos de la especificación VHDL. Por un lado, es posible configurar el número M de secuencias del conjunto a través del parámetro $m = log_2 M$. Además, es posible variar el número N de etapas del correlador y, en consecuencia, la longitud $L = M^N$ de las secuencias a detectar. Asimismo, el número de bits DW con el que se codifica la señal de entrada es también configurable. Por otro lado, de cara a la posterior aplicación del correlador en un sistema práctico en el que los códigos son modulados para ajustarlos a la respuesta en frecuencia del transductor, y la demodulación se realiza de forma asíncrona, se han incluido dos parámetros genéricos adicionales: N_{SM} que indica el número de períodos del símbolo de modulación; y $O_f = \frac{f_a(frec.\ adquisición)}{f_e(frec.\ emisión)}$ que hace referencia al factor de sobremuestreo.

La figura 4.6 muestra la estructura del bloque de mayor jerarquía. Puesto que el número de secuencias del conjunto se define al sintetizar el diseño, el número de puertos de salida del correlador se concreta también en ese momento (se recuerda que el correlador posee M salidas, correspondientes a la correlación de la señal de entrada con las M secuencias del conjunto). Como en VHDL[3] no es posible tener un número de puertos genérico, se ha definido un único puerto de salida de tamaño variable. Este puerto contiene las M salidas del M-ESSC concatenadas, de modo que la correlación $C_{r,s_{i,0_{(N)}}}[\tau]$ con la primera secuencia del conjunto aparece en los $DW + m \cdot N$ bits de mayor peso, y la correlación $C_{r,s_{i,M-1_{(N)}}}[\tau]$ en los bits $DW + m \cdot N$ bits de menor peso.

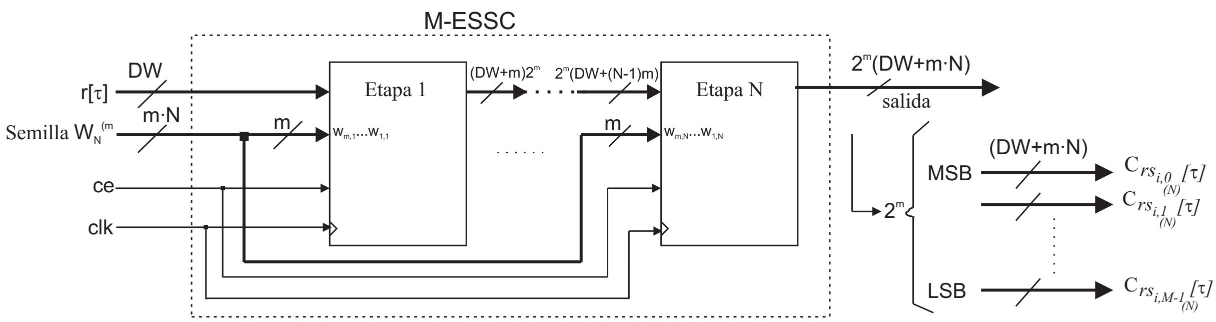


Figura 4.6: Puertos de entrada y salida de la arquitectura propuesta para la implementación del M-ESSC.

Nótese que cada salida $\{C_{r,s_{i,M-1_{(N)}}}[\tau]; 0 \leq i \leq M-1\}$ del correlador requiere $DW+m\cdot N$ bits para evitar el desbordamiento en las operaciones internas del correlador (a razón de m bits adicionales por etapa). Por otro lado, si se tiene en cuenta el efecto de la demodulación, los bits correspondientes a una misma secuencia $s_{i,j}$ se obtienen con una separación de $N_{SM} \cdot O_f$ muestras entre ellos. Es necesario por tanto diezmar la señal demodulada por

[3]VHDL es un lenguaje estandarizado de descripción hardware (*Hardware Description Language*) usado para diseñar circuitos digitales VHSIC (*Very High Speed Integrated Circuits*) y ampliamente utilizado para la programación de dispositivos FPGAs.

dicho factor antes de realizar las correlaciones. El diezmado puede realizarse fácilmente multiplicando los retardos D_n de cada etapa por $N_{SM} \cdot O_f$. Puesto que la anchura del bus de datos se incrementa en cada etapa, la memoria total necesaria puede reducirse situando en las primeras los retardos D_n de mayor valor. Asimismo, sabiendo que los retardos de la primera etapa comparten la entrada al sistema $r[\tau]$ basta con implementar el de mayor valor $(M-1)D_1$, con $D_1 = N_{SM} \cdot O_f \cdot DW \cdot M^{N-1}$ y habilitar el acceso simultáneo a posiciones intermedias. Es más, si la señal recibida va a ser correlada con varios códigos simultáneamente, los correladores pueden compartir este bloque de retardo, según se muestra en la figura 4.7, con el consiguiente ahorro de posiciones de memoria.

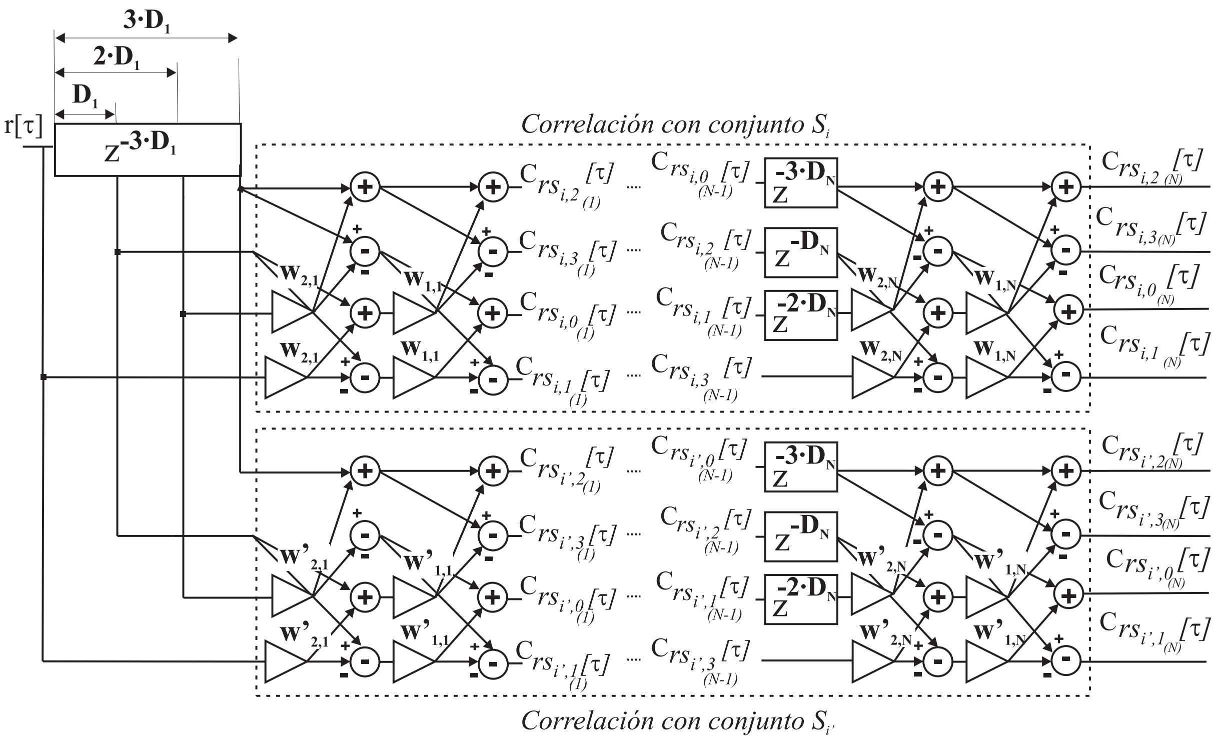


Figura 4.7: Correlación eficiente simultánea de dos conjuntos S_i y $S_{i'}$ con $M = 4$ secuencias complementarias.

Aún sabiendo que en la primera etapa los retardos se configuran de modo distinto, la estructura de todas ellas es similar. Internamente el diseño de las etapas se ha dividido en cuatro bloques, como puede observarse en la figura 4.8.

- El primer bloque implementa los retardos de la etapa. La implementación de los retardos de las etapas internas $(1 < n \leq N)$ en bloques de memoria dedicados SelectRAM resulta un procedimiento costoso, por lo que se han utilizado registros de desplazamiento serie SRL16 de los slices.
- El segundo bloque reordena las salidas retardadas según el algoritmo de reflexión binaria, y no requiere recursos adicionales.
- El bloque combinacional realiza las operaciones de la etapa. Como los coeficientes $w_{k,n}$

tienen valores $\{-1, 1\}$ son realmente éstos los que determinan la configuración final de los sumadores y restadores de cada etapa (véase la figura 4.9). Además, debido a la nueva geometría del correlador, basta con implementar una única sub-etapa y añadir el resto en tiempo de síntesis en función del parámetro m, con lo que se ahorra tiempo de diseño. Por otro lado, para secuencias de longitud elevada el cómputo de las operaciones puede suponer un incremento del retardo combinacional de la etapa. En estos casos conviene añadir registros pipe-line entre sub-etapas para aumentar la frecuencia de funcionamiento del sistema.

- Finalmente, las salidas del bloque anterior son reordenadas según (4.2) para su correcta conexión con la siguiente etapa. Esta conexión se realiza sin necesidad de consumir recursos de la FPGA.

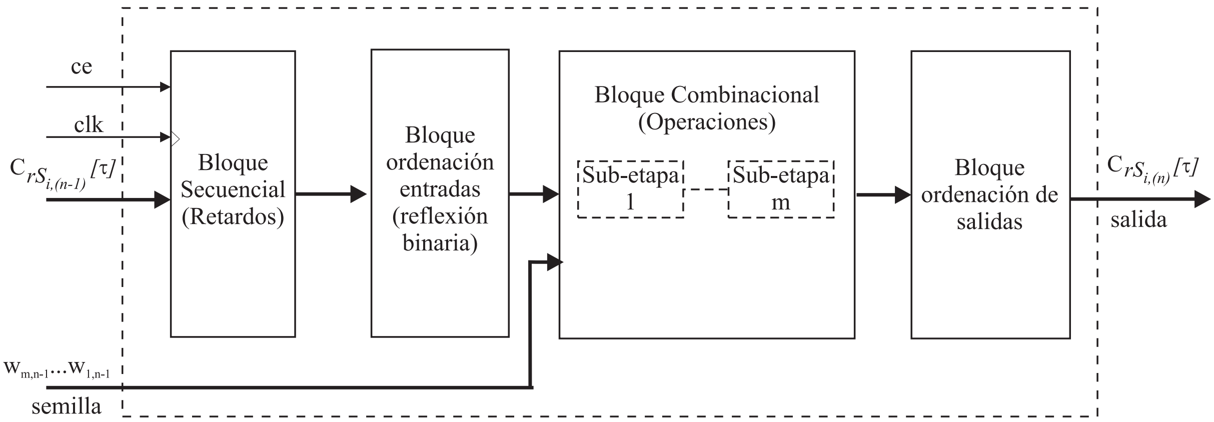


Figura 4.8: Esquema de implementación de una etapa del M-ESSC configurado en pre-síntesis.

Configuración en post-síntesis

Otro posible diseño es aquél en el que el usuario puede cambiar en tiempo de ejecución los parámetros principales del código patrón, esto es, el número $M = 2^m$ de secuencias del conjunto y su longitud $L = M^N$. De este modo, si las condiciones del entorno cambian (aumenta el ruido o el número de emisores simultáneos) el sistema puede adaptarse sin necesidad de pararlo y sintetizar nuevamente el diseño. Para llevar a cabo esta implementación se ha aprovechado que la estructura del correlador de M secuencias (M-ESSC) hace uso del M/2-ESSC, el M/2-ESSC del M/4-ESSC, y así sucesivamente.

Se ha modificado por tanto la estructura descrita en el apartado anterior de forma que m y N sean entradas al diseño. Por otro lado, se ha definido un número M_{max} máximo de secuencias del conjunto y un número N_{max} máximo de etapas del correlador, de modo que el diseño realizado se ha dimensionado para cualquier $M \leq M_{max}$ y $N \leq N_{max}$. La figura 4.10 resume la nueva estructura de una etapa n cualquiera.

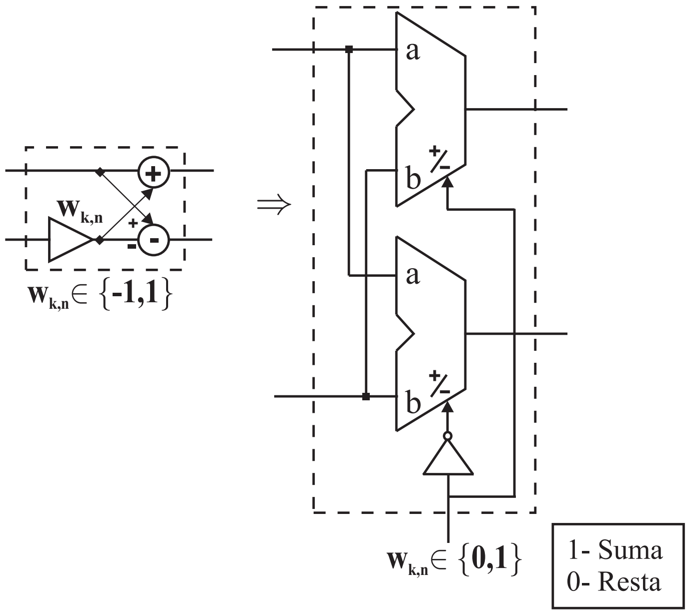


Figura 4.9: Esquema de implementación de un multiplicador, sumador y restador de una etapa del ESSC.

- En el primer bloque se retardan los datos de entrada en función del número M de secuencias elegido. En caso de que el usuario decida trabajar con un número de secuencias $M < M_{max}$, se habilitan sólo las M entradas $\{C_{r,s_{i,j(n)}}[\tau]; M_{max} - M \leq j \leq M_{max} - 1\}$ conectadas a los M retardos de menor valor ($\{(M-1)D_n, \cdots, 1, 0\}$), introduciendo ceros en las $M_{max} - M$ entradas restantes $\{C_{r,s_{i,j(n)}}[\tau]; 0 \leq j \leq M_{max} - M - 1\}$. Además, puesto que estos M retardos de menor valor están dimensionados para un $D_n = N_{SM} \cdot O_f \cdot M_{max}^{(N_{max}-n)}$, es necesario permitir el acceso a los registros de desplazamiento que implementan estos retardos en las posiciones que resultan de utilizar un $D_n = N_{SM} \cdot O_f \cdot M^{(N_{max}-n)}$, teniendo en cuenta que M puede tomar valores en el rango $2 \leq M \leq M_{max}$ y que n es el número de etapa.

- Al igual que en la propuesta de configuración en pre-síntesis, las salidas del bloque anterior son reordenadas según el algoritmo de reflexión binaria.

- El bloque combinacional realiza las operaciones considerando que el valor de los coeficientes $(w_{m_{max}-m,n}, \cdots, w_{1,n})$ es 1, en donde $M_{max} = 2^{m_{max}}$ y $M = 2^m$ con $m_{max}, m \in \mathbb{N} - \{0\}$. De este modo el signo de los datos a procesar en la etapa no se verá afectado por los coeficientes de las sub-etapas $m+1$ a m_{max}. Para simplificar el diseño es el usuario el que pone a 1 los coeficientes no usados.

- El último bloque selecciona de entre las M_{max} posibles salidas las que corresponden a los resultados de correlación con las M secuencias del conjunto elegido. Sabiendo que la salida $x = 0$ de una etapa cualquiera es la de la rama superior y que hay M_{max} salidas por etapa, aquellas salidas x' que contienen los resultados de correlación parcial con las M secuencias del conjunto son $\{0, \frac{M_{max}}{M}, 2 \cdot \frac{M_{max}}{M}, 3 \cdot \frac{M_{max}}{M}, \cdots, (M-1) \cdot \frac{M_{max}}{M}\}$. La relación entre dichas salidas y el índice j que indica el resultado de la correlación parcial $\{C_{r,s_{i,j(n)}}[\tau]; 0 \leq j \leq M-1\}$ obtenido en las mismas es:

$$\begin{array}{ll} x \in \{0, 1, \cdots, M-1\} & x' \subseteq x \in \{0, \frac{M_{max}}{M}, 2 \cdot \frac{M_{max}}{M}, 3 \cdot \frac{M_{max}}{M}, \cdots, (M-1) \cdot \frac{M_{max}}{M}\} \\ \text{Si } x' \cdot \frac{M}{M_{max}} \text{ par:} & j = (M - 2 - x' \cdot \frac{M}{M_{max}}) \\ \text{Si } x' \cdot \frac{M}{M_{max}} \text{ impar:} & j = (M - x' \cdot \frac{M}{M_{max}}) \end{array} \tag{4.3}$$

En la expresión (4.3) x' puede tener valores impares sólo cuando $M = M_{max}$, obteniendo entonces el mismo resultado que en (4.2). Cuando $M < M_{max}$, en las salidas no seleccionadas se obtienen réplicas de los resultados de correlación parcial, pero con distinto signo. En cualquier caso, para evitar que estos valores interfieran en los resultados del siguiente componente conectado al correlador se han puesto a cero.

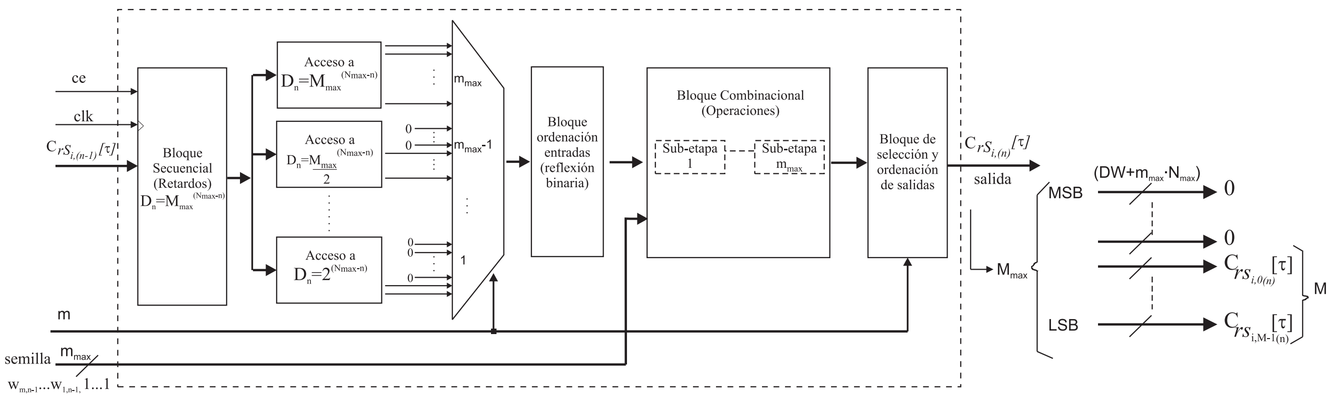


Figura 4.10: Esquema de implementación de una etapa del M-ESSC configurado en post-síntesis.

Se ha fijado también el número N_{max} máximo de etapas del correlador y, utilizando un conjunto de multiplexores, los datos se procesan únicamente en las etapas habilitadas por el usuario en el instante concreto. Específicamente, cuando $N < N_{max}$ las etapas que se inhabilitan son las $N_{max} - N$ primeras, como puede observarse en la figura 4.11.

Este diseño en post-síntesis tiene como ventaja frente al diseño en pre-síntesis que permite cambiar los parámetros del sistema en tiempo de ejecución. Ahora bien, es menos eficiente en cuanto a consumo de recursos y frecuencias de operación, ya que está dimensionado para un número de secuencias máximo M_{max} y una longitud de secuencia máxima $L_{max} = M_{max}^{N_{max}}$, independientemente del número M real de secuencias del conjunto elegido y de su longitud $L = M^N$. Este hecho queda reflejado en la tabla 4.6, donde se ha considerado en primer lugar la emisión de conjuntos de $M = 4$ secuencias complementarias de longitud $L = 64$; y en segundo, la emisión de conjuntos de $M = 2$ secuencias de longitud $L = 2$. Todo ello asumiendo que $M_{max} = 4$, $N_{max} = 3$, que la anchura del bus de datos

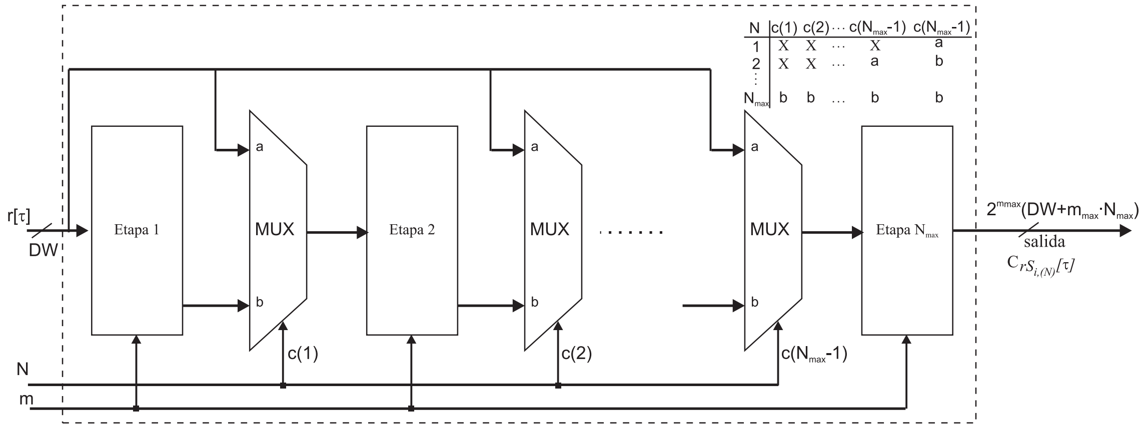


Figura 4.11: Esquema de conexión entre etapas del M-ESSC configurable en post-síntesis.

de entrada es de $DW = 8$ bits, y sin tener en cuenta el efecto de la demodulación[4]. Dado que sintetizar el diseño nuevamente no es un proceso costoso, que el diseño en pre-síntesis es más óptimo desde el punto de vista de la implementación (incluso cuando $M = M_{max}$ y $N = N_{max}$) y que el correlador directo con el que se va a comparar es también configurable en post-síntesis, se ha tomado como referencia este diseño para los siguientes apartados. No obstante, cabe mencionar que en ambientes exteriores en donde el entorno es menos estable el diseño en post-síntesis puede ser de gran utilidad, ya que puede adaptarse al medio en tiempo de ejecución.

xc2vp100	$M = 4,\ L = 64$		$M = 2,\ L = 2$	
	Pre-síntesis	Post-síntesis	Pre-síntesis	Post-síntesis
Slices	372	582	27	582
LUTs	642	914	44	914
SRL16	90	144	8	144
IOBs	72	76	29	76
freq. max. f_{FPGA}	117.233 MHz	101.410 MHz	130.446 MHz	101.410 MHz

Tabla 4.6: Comparativa de recursos y tiempos de ejecución requeridos por las implementaciones en pre-síntesis y post-síntesis del M-ESSC.

Las ventajas de una implementación eficiente frente a una directa quedan manifiestas en la tabla 4.7, en donde se compara el rendimiento del correlador M-ESSC configurado

[4]Los resultados mostrados en este capítulo corresponden a un caso general en el que sólo se tienen en cuenta el número de secuencias del código, su longitud y el tamaño de los datos de entrada. El resto de parámetros, específicos de la modulación empleada y por tanto dependientes de la aplicación final, no se consideran ($N_{SM} = O_f = 1$).

en post-síntesis, con el correlador directo serie representado en la figura 4.1.b, y también configurado en post-síntesis. Ambos correladores realizan la correlación simultánea de la señal de entrada con las $M = 4$ secuencias complementarias de longitud $L = 64$ de un conjunto, considerando que la señal de entrada tiene un ancho de bus de datos de $DW = 8$ bits. Si bien el M-ESSC requiere un mayor número de *slices*, obtiene un nuevo resultado de correlación cada 8.53 ns, frente a los 480 ns que necesita el correlador directo serie. Esta diferencia es debida a que el M-ESSC proporciona un nuevo resultado con la frecuencia de reloj de la FPGA ($f_{FPGA} = f_H = f_L$), mientras que el directo lo hace con $f_L = \frac{f_{FPGA}=f_H}{L}$.

xc2vp100	M=4, L=64	
	4-ESSC	Directo
Slices	372	93
LUTs	642	141
SRL16	90	0
Bloques RAM	0	6
IOBs	72	67
freq. max. f_{FPGA}	117.233 MHz	131.182 MHz
Rendimiento (Mmuestras/s)	117.233	2.05
Tiempo de correlación	8.53 ns	487.87 ns

Tabla 4.7: Comparativa de recursos y tiempos de ejecución empleando un ESSC y correlador directo serie para la detección de un conjunto de $M = 4$ secuencias complementarias de longitud $L = 64$.

4.2.2. Adaptación a emisión mediante macro-secuencias

En la sección 3.3 del capítulo anterior quedó demostrado que la ACF y CCF de una macro-secuencia cualquiera de CSS puede expresarse en función de la suma de correlaciones con las secuencias del conjunto que la componen. Así, la implementación propuesta del ESSC puede ser adaptada fácilmente al esquema de emisión mediante macro-secuencias.

Cuando la macro-secuencia se obtiene a partir de la concatenación de las M secuencias de un conjunto S_i, puede utilizarse el ESSC sin ninguna modificación obteniendo en cada rama del mismo la correlación de la macro-secuencia Msc_i con cada secuencia $s_{i,j}$ del conjunto. Como las secuencias $\{s_{i,j}; 0 \leq j \leq M-1\}$ se encuentran separadas L muestras entre ellas, las ACF $C_{s_{i,j},s_{i,j}}$ de cada una de ellas se obtienen en las salidas del ESSC con un desfase de L muestras. Asimismo, el ruido auto-inducido $C_{s_{i,j},s_{i,h}}$ y $C_{s_{i,j+1},s_{i,h+1}}$, con $j \neq h$ y $0 \leq j, h \leq M-1$, consecuencia del proceso de concatenación también se encuentra desfasado L muestras. Por tanto, añadiendo a la salida del ESSC $M-1$ etapas de retardo según (4.4) se obtiene la correlación de la señal de entrada $r[\tau]$ con la macro-secuencia Msc_i. La estructura final de este nuevo correlador, denominado Msc-ESSC, puede verse en la figura 4.12.

$$C_{r,Msc_i}[\tau] = C_{r,s_{i,0}}[\tau-(M-1)\cdot D_c]+C_{r,s_{i,1}}[\tau-(M-2)\cdot D_c]+\cdots+C_{r,s_{i,M-2}}[\tau-D_c]+C_{r,s_{i,M-1}}[\tau] \tag{4.4}$$

Donde $D_c = L$ es un retardo que tiene en cuenta la separación de L muestras entre dos secuencias consecutivas de una macro-secuencia, para poder realizar la suma en fase de las correlaciones. Además, si se considera el efecto de la demodulación, este retardo dependerá también del número de períodos del símbolo de modulación N_{SM}, y del factor de sobremuestreo O_f, con lo que queda: $D_c = N_{SM} \cdot O_f \cdot L$.

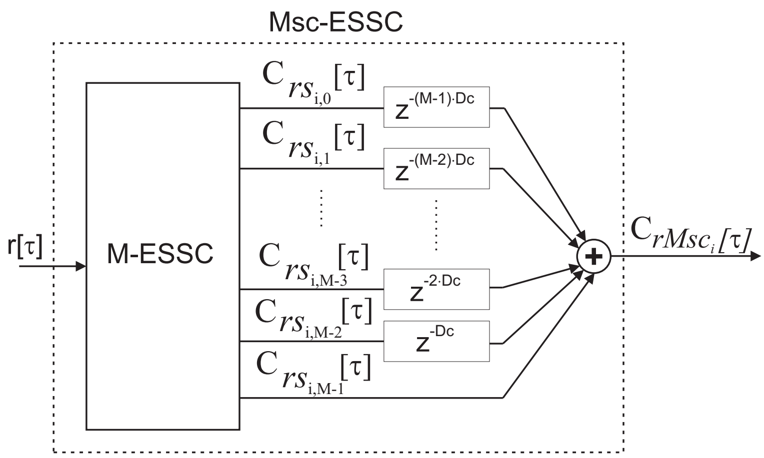


Figura 4.12: Diagrama de bloques de un correlador de macro-secuencias (Msc-ESSC) utilizando un ESSC.

Cuando la macro-secuencia se genera a partir del entrelazado de los bits de las secuencias $s_{i,j}$, los bits correspondientes a una misma secuencia se encuentran separados M muestras. Sabiendo esto, y teniendo en cuenta también el efecto de la demodulación, los retardos D_n internos del ESSC deben ir multiplicados por $N_{SM}\cdot O_f\cdot M$, es decir, $D_n = N_{SM}\cdot O_f\cdot M^{N-n+1}$. Asimismo, los resultados de las auto-correlaciones de las secuencias del conjunto se obtienen con un desfase de $D_e = N_{SM} \cdot O_f$ muestras entre cada rama del ESSC y la rama inferior, de modo que es necesario añadir $M-1$ etapas de retardo adicionales a la salida del ESSC para sumarlos en fase. Tras la suma, se tiene la correlación de la señal de entrada con la macro-secuencia Mse_i. La estructura de este correlador, denotado como Mse-ESSC, es similar a la mostrada en la figura 4.12: basta con sustituir los retardos D_c a la salida del ESSC, por los retardos propios del entrelazado D_e, y tener en cuenta que los retardos internos del ESSC van multiplicados por $N_{SM} \cdot O_f \cdot M$.

Las tablas 4.8 y 4.9 muestran los recursos consumidos por macro-secuencias de longitud

$L_{Ms} = 64$ obtenidas mediante concatenación y entrelazado, respectivamente. Las macro-secuencias se han generado a partir de 2-CSS de longitud $L = 32$, 4-CSS de longitud $L = 16$ y 8-CSS de longitud $L = 8$. La tabla 4.10 recoge los recursos empleados y frecuencias de operación en caso de usar macro-secuencias de longitud $L_{Ms} = 256$, obtenidas a partir de 2-CSS de longitud $L = 128$, 4-CSS de longitud $L = 64$ y 16-CSS de longitud $L = 16$. En las tres tablas se ha considerado que $N_{SM} = O_f = 1$, y que el ancho de bus de datos de entrada al correlador es $DW = 8$ bits. Los correladores se han implementado en una Virtex-II Pro de Xilinx, y los resultados se proporcionan después del emplazamiento y ruteado.

xc2vp100	Msc(2,32)	Msc(4,16)	Msc(8,8)
Slices	196	319	525
LUTs	310	502	800
SRL16	76	126	232
IOBs	29	28	27
freq. max. f_{FPGA}	114.286 MHz	87.336 MHz	76.652 MHz

Tabla 4.8: Comparativa de recursos y tiempos de ejecución para la correlación de macro-secuencias de longitud $L_{Ms} = 64$ obtenidas mediante concatenación.

xc2vp100	Mse(2,32)	Mse(4,16)	Mse(8,8)
Slices	191	283	426
LUTs	305	466	701
SRL16	71	90	133
IOBs	29	28	27
freq. max. f_{FPGA}	113.135 MHz	89.799 MHz	71.459 MHz

Tabla 4.9: Comparativa de recursos y tiempos de ejecución para la correlación de macro-secuencias de longitud $L_{Ms} = 64$ obtenidas mediante entrelazado.

En las tablas 4.8 a 4.10 puede observarse como para macro-secuencias de la misma longitud, ya sean generadas mediante concatenación (Msc) o entrelazado (Mse), los mejores resultados se obtienen cuando se usan aquellas que han sido generadas con el menor número M de secuencias posible. En cuanto a la frecuencia de operación, cuantas menos secuencias M tenga el conjunto, menos costoso resulta realizar la suma combinacional de las M ramas de salida del ESSC, por lo que la frecuencia de trabajo aumenta. Por otro lado, la cantidad total de elementos de memoria a utilizar disminuye si se elige $M = 2$. En caso de utilizar correladores de una única etapa, esto es con $M = L = 2^m$, se tiene que el número de bits N_B a almacenar es (4.5) para concatenación y (4.6) para entrelazado.

xc2vp100	Ms(2,128)		Ms(4,64)		Ms(16,16)	
	Concat.	Entrel.	Concat.	Entrel.	Concat.	Entrel.
Slices	410	365	732	540	2356	1096
LUTs	582	537	1024	832	3110	1850
SRL16	230	185	426	234	1560	300
IOBs	33	33	32	32	30	30
freq. max. f_{FPGA}	125.502 MHz	115.447 MHz	88.067 MHz	86.648 MHz	71.490 MHz	66.903 MHz

Tabla 4.10: Comparativa de recursos y tiempos de ejecución para la correlación de macro-secuencias de longitud $L_{Ms} = 256$.

$$N_{B_{Msc(M,M)}} = N_{SM} \cdot O_f \cdot (M-1) \cdot DW + N_{SM} \cdot O_f \cdot \frac{M^2(M-1)}{2} \cdot (DW + m) \quad (4.5)$$

$$N_{B_{Mse(M,M)}} = N_{SM} \cdot O_f \cdot M(M-1) \cdot DW + N_{SM} \cdot O_f \cdot \frac{M(M-1)}{2} \cdot (DW + m) \quad (4.6)$$

Eligiendo macro-secuencias de la misma longitud $L_{Ms} = M^2$ que las anteriores, pero obtenidas a partir de CSS con el menor número de secuencias posible, esto es con $M' = 2$ y $L' = \frac{M^2}{2}$, disminuye el número de bits a almacenar, siendo (4.7) para concatenación y (4.8) para entrelazado.

$$N_{B_{Msc(2,\frac{M^2}{2})}} = N_{SM} \cdot O_f \cdot \frac{M^2}{4} \left[\frac{2M^2-4}{M^2} DW + \sum_{i=1}^{2m-2} \frac{i}{2^i} \right] + N_{SM} \cdot O_f \cdot \frac{M^2}{2} (DW + 2m - 1) \quad (4.7)$$

$$N_{B_{Mse(2,\frac{M^2}{2})}} = N_{SM} \cdot O_f \cdot \frac{M^2}{2} \left[\frac{2M^2-4}{M^2} DW + \sum_{i=1}^{2m-2} \frac{i}{2^i} \right] + N_{SM} \cdot O_f \cdot (DW + 2m - 1) \quad (4.8)$$

Para mayor claridad, se han representado gráficamente en la figura 4.13 los resultados obtenidos con $Ms(M, M)$ y $Ms(2, \frac{M^2}{2})$. Además de las ventajas de usar macro-secuencias obtenidas a partir de pares Golay (2-CSS), se observa que el correlador Mse-ESSC de macro-secuencias generadas mediante entrelazado requiere menos recursos que el Msc-ESSC, adaptado a macro-secuencias generadas mediante concatenación; aunque la frecuencia de operación es ligeramente menor. Por otro lado, comparando las tablas 4.8 y 4.9 con la tabla 4.10 puede verificarse que macro-secuencias de mayor longitud consumen más recursos.

Finalmente, la tabla 4.11 compara las necesidades hardware de los correladores Msc-ESSC y Mse-ESSC con un correlador directo serie, cuando se han empleado macro-secuencias de longitud $L_{Ms} = 256$ obtenidas a partir de parejas Golay, y un ancho de

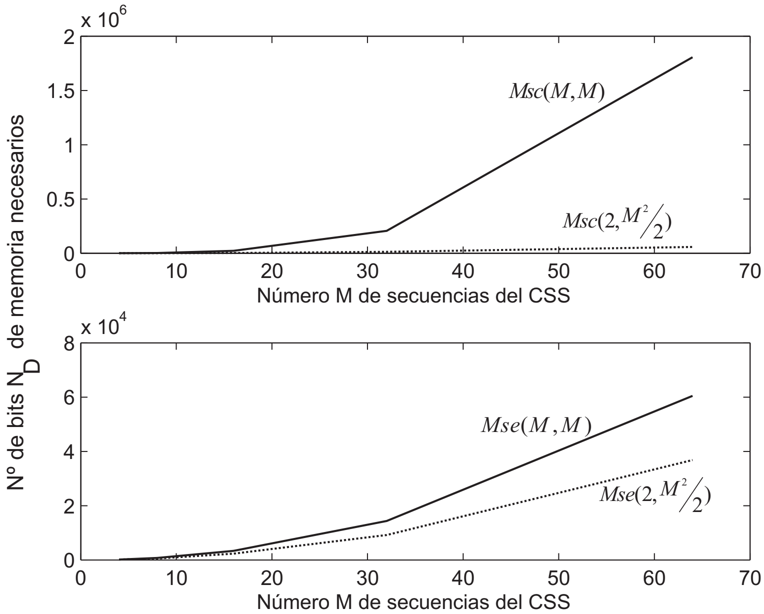


Figura 4.13: Número de bits de memoria necesarios para la correlación de macro-secuencias $Ms(M, M)$ y $Ms(2, \frac{M^2}{2})$ empleando correladores basados en el ESSC.

bus de datos de entrada $DW = 8$. Como era de esperar, se demuestra que para trabajar en tiempo real son más adecuados los correladores Msc-ESSC y Mse-ESSC.

xc2vp100	Ms(2,128)		
	Msc-ESSC	Mse-ESSC	Directo
Slices	410	365	71
LUTs	582	537	117
SRL16	230	185	0
Bloques RAM	0	0	3
IOBs	33	33	27
freq. max. f_{FPGA}	125.502 MHz	115.447 MHz	139.509 MHz
Rendimiento (Mmuestras/s)	125.502	115.447	0.55
Tiempo de correlación	7.97 ns	8.66 ns	1835 ns

Tabla 4.11: Estudio comparativo de los resultados de implementación empleando correladores basados en el ESSC y un correlador directo, para la detección de macro-secuencias de longitud $L_{Ms} = 256$.

4.3. Propuesta de un algoritmo de correlación eficiente de códigos LS y resultados de implementación

En esta sección se aprovecha la relación existente entre los códigos LS y los CSS para proponer algoritmos de generación y correlación eficientes de códigos LS que reduzcan el número de operaciones a realizar, permitiendo el procesamiento en tiempo real y el uso de secuencias más largas. Así, a partir de los algoritmos eficientes de generación y correlación de CSS (ESSG y ESSC respectivamente) se han desarrollado los nuevos algoritmos propuestos para códigos LS, denotados como ELSG (generador eficiente de códigos LS) y ELSC (correlador eficiente de códigos LS).

Se explican a continuación los algoritmos eficientes propuestos para los dos esquemas de generación de códigos LS evaluados en esta tesis (se recomienda revisar la sección 2.2.6 en donde se encuentran resumidos ambos métodos).

4.3.1. Algoritmos eficientes asociados a códigos LS generados a partir de parejas Golay

El correlador eficiente de parejas Golay desarrollado por Budisin y Popovic [Bud91, Pop99] se deriva de un algoritmo recursivo que permite generar parejas Golay partiendo de las secuencias elementales delta $\delta[\tau]$. Este algoritmo de generación, denotado como Generador Eficiente Golay (EGG) tiene la misma estructura que el 2-ESSG propuesto por De Marziani et al. [DMUH+07b], y puede utilizarse para la generación de los dos pares incorrelados S_0 y S_1 que componen los códigos LS propuestos por Stańczak et al. [SBH01]. Con esto, el algoritmo ELSG puede obtenerse siguiendo los pasos que se detallan a continuación, cuya representación gráfica aparece en la figura 4.14:

- Paso 1: Obtención de los dos pares Golay incorrelados $S_0 = (s_{0,0}, s_{0,1})$ y $S_1 = (s_{1,0}, s_{1,1})$ mediante el uso de dos generadores eficientes Golay (recuérdese la arquitectura del 2-ESSG en la figura 4.2.a). Para que los pares sean incorrelados las semillas de generación deben ser $Wd_N^{(1}$ y $Wd_N'^{(1} = (Wd_N^{(1} + L_0/2) \bmod L_0$, en donde $L_0 = 2^N$ es la longitud de los pares Golay y $N \in \mathbb{N} - \{0\}$ es el número de etapas del generador Golay.

- Paso 2: Un conjunto de K multiplexores, donde $K = 2^n, n \in \mathbb{N}$ es el número de códigos LS de la familia, determina el retardo a aplicar a cada secuencia de los pares Golay.

  Los $K/2$ multiplexores inferiores (integrados en el bloque 3), son los encargados de retardar las secuencias 0 de cada par S_0 y S_1. La elección de la secuencia $s_{0,0}$ o $s_{1,0}$ en cada multiplexor depende del valor del vector $\Pi = [\pi_0, \pi_1, \cdots, \pi_{\frac{K}{2}}] \in \{0, 1\}$, de modo que cuando $\pi_i = 0$ se retarda la secuencia $s_{0,0}$ y cuando $\pi_i = 1$ la otra. Los retardos a aplicar en este bloque tienen valores en el rango $\{0,\ L_0,\ 2\cdot L_0, \cdots, (K/2-1)\cdot L_0\}$. Esto es, conforman la base de lo que será la primera mitad del código LS.

Los $K/2$ multiplexores superiores (pertenecientes al bloque 2) retardan las secuencias $s_{0,1}$ y $s_{1,1}$ de los pares Golay S_0 y S_1 respectivamente. Estas secuencias formarán la segunda mitad del código LS; obsérvese que los retardos ahora están en el rango $\{(K/2)\cdot L_0+W,\ (K/2+1)\cdot L_0+W,\ \cdots,\ (K-1)\cdot L_0+W\}$; el retardo $W \leq L_0-1$ corresponde al intervalo de ceros a insertar en la mitad del código. Los multiplexores de este bloque están controlados por el mismo vector Π del bloque 3.

- Paso 3: Las secuencias Golay retardadas de salida del paso 2 se multiplican por los elementos $h_{k,i}$ de la columna k de una matriz de Hadamard de tamaño $(\frac{K}{2}\times\frac{K}{2})$ y los valores resultantes se suman según (4.9) para obtener el código LS final de longitud $L = KL_0 + W$.

$$G_k[z] = \sum_{i=0}^{K/2-1} h_{k,i}\cdot z^{-i\cdot L_0}\cdot[S_{\pi_i,0}[z] + z^{-(\frac{K}{2}L_0+W)}S_{\pi_i,1}[z]] \tag{4.9}$$

En (4.9) $G_k[z]$ es la transformada Z del código LS $g_k(\tau)$, el resto de parámetros se encuentran definidos en la sección 2.2.6, aunque se ha hecho un breve repaso en los párrafos previos. Por otro lado, si se sustituye en (4.9), y por tanto también en el esquema de la figura 4.14, el vector Π por su complementario Π^* se obtienen los siguientes $K/2$ códigos de la familia LS de K códigos.

Modificando el algoritmo de generación para generar las secuencias Golay reflejadas en el tiempo $(s_{0,0}[L_0-1-\tau], s_{0,1}[L_0-1-\tau])$ y $(s_{1,0}[L_0-1-\tau], s_{1,1}[L_0-1-\tau])$ y permutando el orden de los retardos en (4.9), se obtiene un filtro correlador acoplado a los códigos LS, que puede expresarse según (4.10) y que se ha denominado correlador eficiente de códigos LS (ELSC) por requerir menos operaciones que un correlador directo.

$$\begin{aligned} C_{R,G_k}[z] &= \textstyle\sum_{i=0}^{K/2-1} h_{k,i} z^{-(\frac{K}{2}-i-1)\cdot L_0}\left[z^{-(\frac{K}{2}\cdot L_0+W)}C_{R,S_{\pi_i,0}}[z] + C_{R,S_{\pi_i,1}}[z]\right] \\ C_{R,G_{k+K/2}}[z] &= \textstyle\sum_{i=0}^{K/2-1} h_{k,i} z^{-(\frac{K}{2}-i-1)\cdot L_0}\left[z^{-(\frac{K}{2}L_0+W)}C_{R,S_{\pi_i^*,0}}[z] + C_{R,S_{\pi_i^*,1}}[z]\right] \end{aligned} \tag{4.10}$$

En donde $C_{R,G_k}[z]$, con $0 \leq k \leq \frac{K}{2}$, representa la transformada Z de la correlación entre la señal de entrada $r[\tau]$ y el código LS $g_k[\tau]$; del mismo modo, $C_{R,S_{\pi_i,0}}(z)$ y $C_{R,S_{\pi_i,1}}(z)$ representan la transformada Z de la correlación entre la señal de entrada $r[\tau]$ y las secuencias 0 y 1 de los conjuntos S_0 y S_1. Estas últimas correlaciones pueden llevarse a cabo mediante el correlador eficiente Golay (EGC o 2-ESSC), lo que redunda en una reducción del número total de operaciones a realizar en la correlación de códigos LS. La implementación de este correlador ELSC puede dividirse en tres pasos de forma similar a como se hizo en el generador ELSG (véase la figura 4.15):

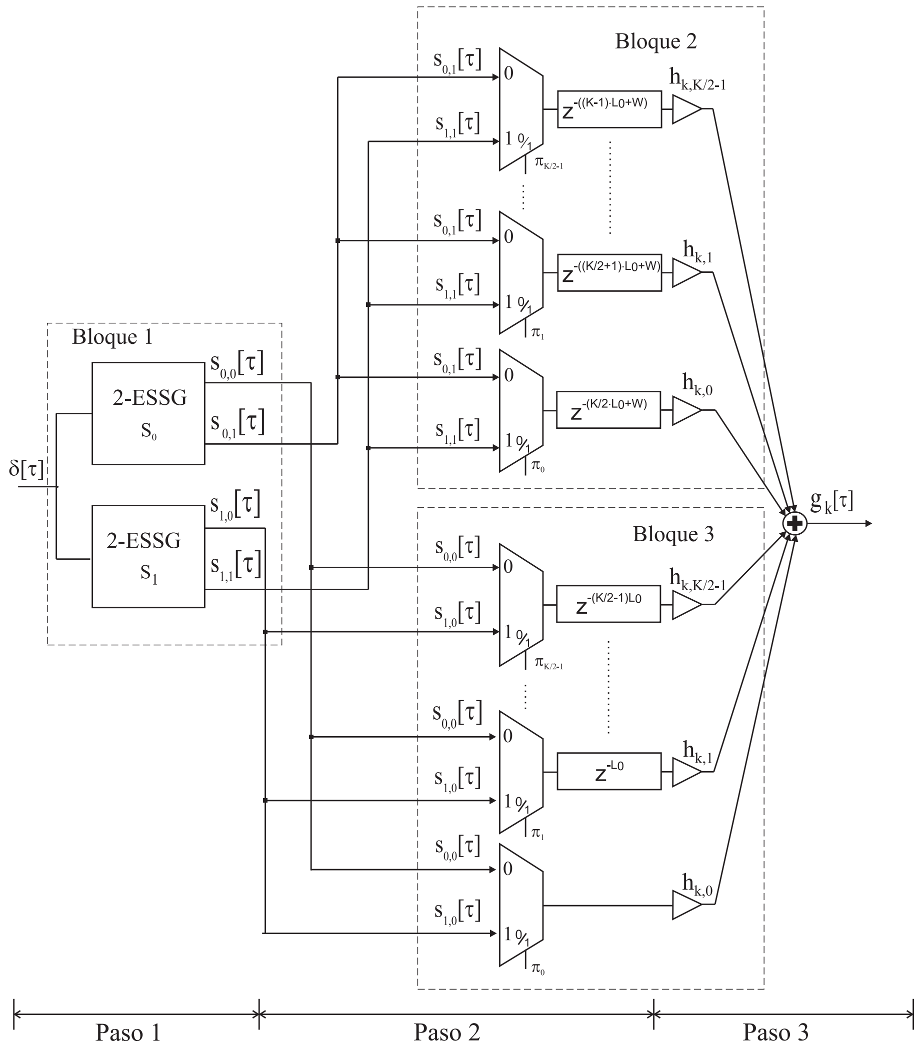


Figura 4.14: Diagrama de bloques de un generador eficiente de códigos LS (ELSG) generados según [SBH01].

- Paso 1: Correlación de la señal de entrada con los dos pares S_0 y S_1 Golay incorrelados que componen el código LS. Estas correlaciones se realizan de modo eficiente gracias a dos 2-ESSC. Cada 2-ESSC realiza la correlación de la secuencia de entrada con las dos secuencias del par, de modo que a la salida del paso 1 están disponibles las correlaciones $(C_{r,s_{0,0}}, C_{r,s_{0,1}})$ y $(C_{r,s_{1,0}}, C_{r,s_{1,1}})$.

- Paso 2: Las correlaciones obtenidas se retardan dependiendo del vector Π o Π^* con el que se generó el código a detectar y en orden inverso a como se hizo en la generación. Nótese que los retardos de mayor valor se aplican ahora a la correlación de la señal de entrada con las secuencias 0 de cada conjunto, y que los coeficientes π_i determinan los retardos $(K-i-1)L_0+W$ a aplicar en el bloque 2. En el bloque 3, sin embargo, los retardos son $(K/2-1-i)L_0$ y se aplican sobre las correlaciones con las secuencias 1 de cada par Golay.

- Paso 3: Las correlaciones retardadas se multiplican por los elementos $h_{k,i}$ de la columna k de la matriz de Hadamard usada en la generación y se suman. Como los elementos $h_{k,i} \in \{-1, 1\}$, las multiplicaciones se reducen a sumas y restas. En definitiva $h_{k,i}$ indica el signo con el que se concatenan los resultados de las correlaciones C_{r,S_0} y C_{r,S_1}, y el vector Π el orden en el que se concatenan dichas correlaciones para obtener el resultado final $C_{r,g_k}[\tau]$, esto es, la correlación de la señal de entrada $r[\tau]$ con el código LS $g_k[\tau]$.

La tabla 4.12 muestra una comparativa entre el número de operaciones y necesidades de memoria del ELSC propuesto y un correlador directo serie convencional. Según se ha adelantado, en ambos correladores las operaciones de multiplicación pueden reducirse a sumas y restas. En el cálculo del número de bits de memoria necesarios para almacenar los datos de la correlación, se han tenido en cuenta parámetros específicos de la aplicación, como el ancho de bus de datos DW de la señal de entrada, el factor de sobremuestreo O_f utilizado cuando la señal es adquirida y el número N_{SM} de períodos del símbolo de modulación. Por otro lado, obsérvese que en el correlador directo no se computan como operaciones las multiplicaciones y sumas con los W valores nulos del código. Los resultados de la tabla muestran que el ELSC reduce en gran medida el número de operaciones a realizar en la correlación, a costa de emplear más memoria que el correlador directo. Puesto que en términos de implementación hardware la reducción en el número de operaciones conlleva más ventajas que el incremento de memoria asociado, se considera la implementación basada en el ELSC más óptima que la basada en un correlador directo.

Resultados de implementación del ELSC propuesto

En la implementación hardware del correlador ELSC se han considerado una serie de parámetros genéricos de la especificación VHDL. Estos parámetros se determinan al sintetizar el diseño y permiten configurar el número de etapas N de los correladores Golay,

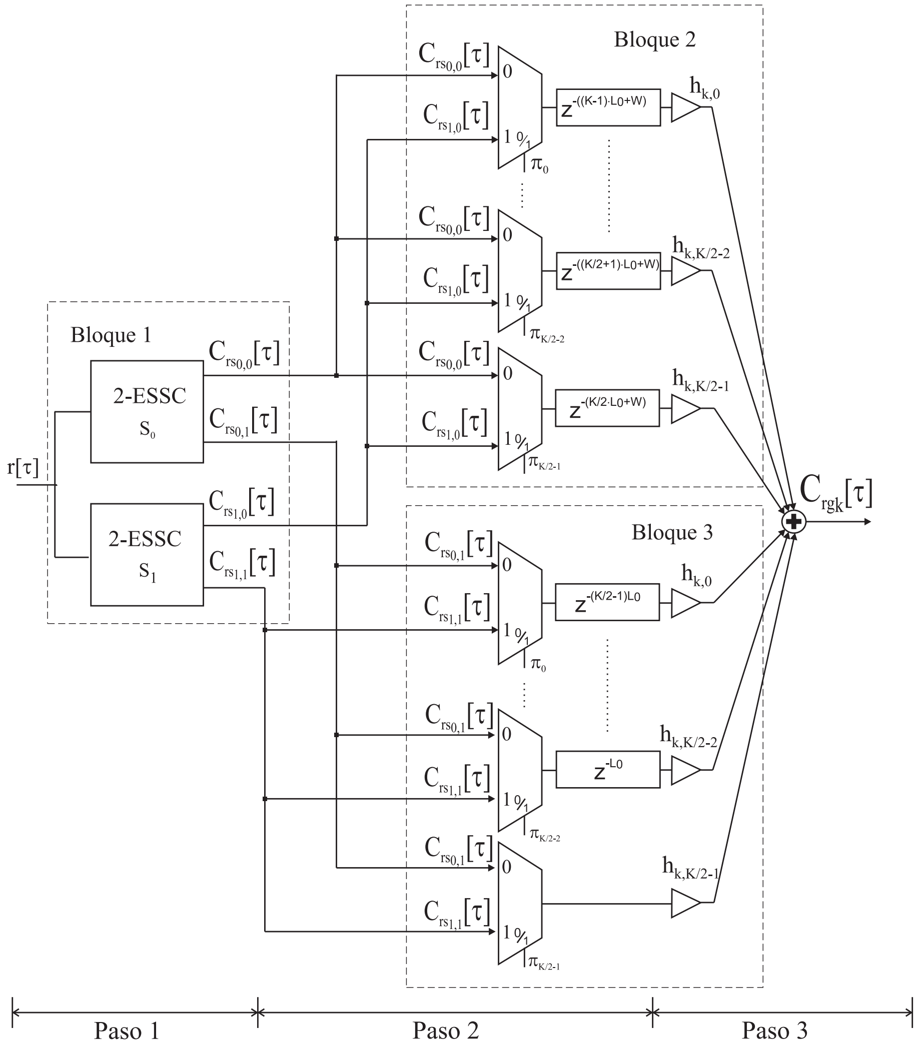


Figura 4.15: Diagrama de bloques de un correlador eficiente de códigos LS (ELSC) generados según [SBH01].

Implementación	Correlador directo serie	ELSC	
Productos	$K \cdot 2^N$	Paso 1	$2N$
		Paso 3	K
		TOTAL	$2N + K$
Sumas	$K \cdot 2^N - 1$	Paso 1	$4N$
		Paso 3	$K - 1$
		TOTAL	$4N + K - 1$
Bits de memoria	$(N_{SM} \cdot O_f \cdot DW + 2)(K \cdot 2^N + W)$ $+(DW + Nlog_2 K)$	Paso 1	$2N_{SM} \cdot O_f \left[DW(2^N - 1) + 2^N \sum_{i=1}^{N-1} \frac{i}{2^{i+1}}\right]$
		Paso 2	$N_{SM} \cdot O_f \cdot (DW + N) \cdot \left[\frac{K(2^N(K-1)+W)}{2}\right]$
		TOTAL	$N_{SM} \cdot O_f \left[2DW(2^N - 1) + 2^{N+1} \sum_{i=1}^{N-1} \frac{i}{2^{i+1}} + \right.$ $\left. +(DW + N) \cdot \left(\frac{K(2^N(K-1)+W)}{2}\right)\right]$

Tabla 4.12: Comparativa de las necesidades computacionales de un correlador directo serie y el ELSC propuesto para la correlación de códigos LS de longitud $L = K \cdot 2^N + W$, generados según [SBH01].

y por tanto la longitud $L_0 = 2^N$ de los pares Golay que componen el código LS. Además, puesto que los componentes del bloque 2 y 3 (véase la figura 4.15) dependen del número K de códigos de la familia, este parámetro es también configurable a través de n, siendo $K = 2^n, n \in \mathbb{N}$. Por otro lado, se puede modificar el número de ceros W en función de los disponibles en el código LS a detectar, y teniendo en cuenta que $W \leq L_0 - 1$. Este número de ceros es el que va a determinar el tamaño $2W + 1$ de la IFW. Otros parámetros configurables son el ancho de bus de datos DW, el número N_{SM} de períodos del símbolo de modulación y el factor de sobremuestreo O_f.

En la figura 4.16 se muestran los puertos de entrada y salida del ELSC. Del total de entradas, cuatro están dedicadas a configuración del código g_k con el que se desea realizar la correlación. Así, el usuario debe introducir las semillas de los pares Golay incorrelados S_0 y S_1 que conforman el código LS, el vector Π y la columna k de la matriz de Hadamard utilizados en la generación del código LS que desea detectar en la señal recibida. Las otras entradas del diseño corresponden a la señal recibida $r[\tau]$, al reloj del sistema clk y a un habilitador de dicho reloj ce.

La arquitectura interna del ELSC puede observarse en la figura 4.17. Se basa en seis módulos distintos que implementan el diagrama de bloques descrito en la figura 4.15. El primer módulo *2-ESSC* corresponde al correlador eficiente Golay [Bud91, Pop99] y el segundo *MUX* implementa los $K/2$ multiplexores de los bloques 2 y 3. El tercer módulo contiene $K/2$ bloques de retardo, todos de valor $N_SM \cdot O_f \cdot (DW + N)(K/2L_0 + W)$ y el cuarto los retardos $N_{SM} \cdot O_f \cdot (DW + N) \cdot \{L_0,\ 2L_0,\ \cdots,\ (K/2 - 1)L_0\}$. Estos retardos han sido implementados en registos de desplazamiento SRL16 de la FPGA. El módulo *Hadamard* determina la polaridad de las salidas retardadas de los correladores y el último módulo suma las salidas de los dos bloques de Hadamard para obtener el resultado de correlación C_{r,g_k}.

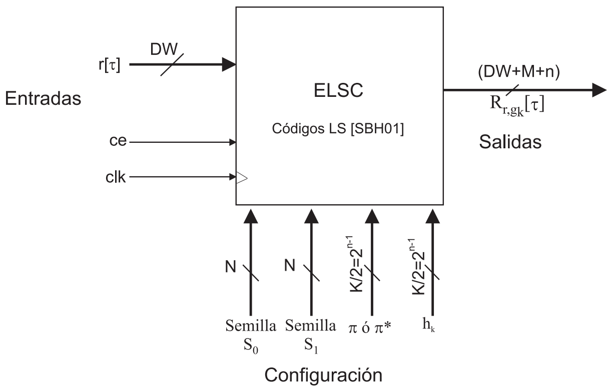


Figura 4.16: Puertos de entrada y salida del ELSC propuesto para códigos LS [SBH01].

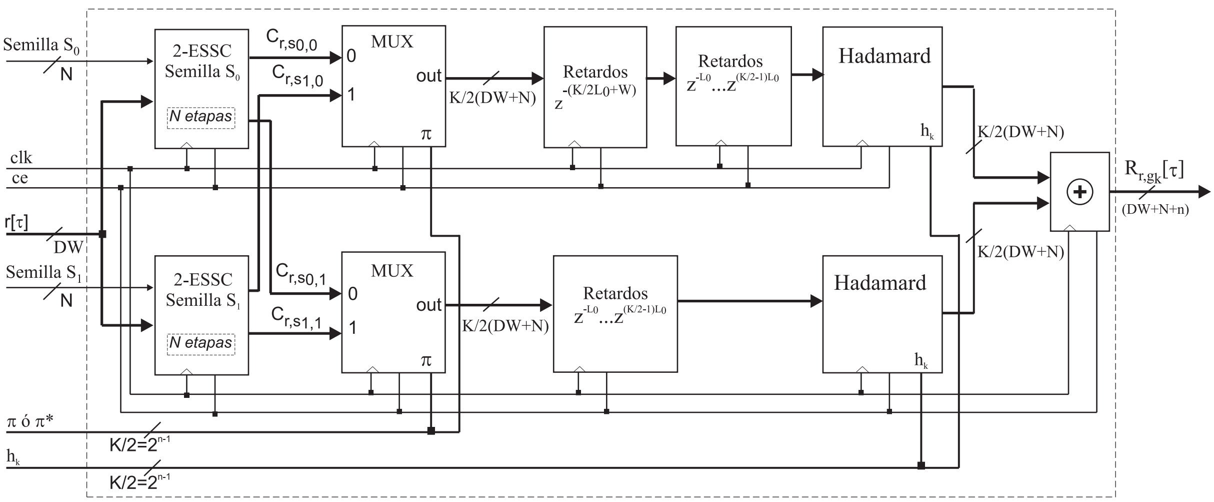


Figura 4.17: Esquema de implementación del ELSC propuesto para códigos LS [SBH01].

El ELSC descrito se ha implementado en una FPGA xc2vp100 de Xilinx [Xil07] ocupando, en función de los parámetros $LS(2, L_0, K)$, los recursos que se especifican en la tabla 4.13. Esta tabla se ha obtenido considerando $DW = 8$, $N_{SM} = O_f = 1$, $W = L_0 - 1$ y la correlación de un único código de los K disponibles en la familia. Las columnas segunda y tercera corresponden a los resultados de implementación de dos códigos de longitud $L = 79$ y $L = 159$ bits pertenecientes a familias con $K = 4$ códigos. La diferencia de longitud de ambos códigos se debe al uso de parejas Golay de distinto tamaño. Como es de esperar, si se utilizan parejas Golay de mayor longitud y por tanto un código LS mayor se necesitan más recursos en la FPGA. Se observa, sin embargo, un ligero aumento de la frecuencia de operación cuando el código es más largo. Este aumento puede justificarse por un mayor esfuerzo de las herramientas de implementación en el mapeado y ruteado de los componentes que haya redundado en una reducción del desfase de reloj (*clock skew*). Las dos siguientes columnas de la tabla corresponden a los resultados de correlación con códigos de familias con $K = 8$ códigos; y las dos últimas corresponden al caso de códigos pertenecientes a familias con $K = 16$. Los códigos elegidos tienen longitudes similares a los de las columnas segunda y tercera, pudiéndose observar que un aumento de K implica un mayor consumo de recursos y una disminución de las frecuencias de operación, aún cuando la longitud de los códigos es parecida. Si se consideran dos códigos LS $g_k[\tau]$ y $g'_k[\tau]$ con valores $LS(2, L_0, K)$ y $LS'(2, \frac{L_0}{2}, 2 \cdot K)$ y $W = L_0 - 1$, el primero de ellos necesita dos 2-ESSCs con una etapa más de correlación, pero reduce a la mitad el número de multiplexores a utilizar y necesita $N_{SM} \cdot O_f \cdot (DW + N) \cdot (\frac{L_0K^2 - K}{2})$ bits de memoria para retrasar los resultados de las correlaciones con los pares Golay; mientras que el segundo requiere el doble de multiplexores y $N_{SM} \cdot O_f \cdot (DW + N) \cdot (L_0K^2 - K)$ bits de memoria. Además, en el primer caso deben sumarse las K ramas de salida del paso 2 en un sólo ciclo de reloj, mientras que en el segundo la suma combinacional es de $2K$ ramas y por tanto más costosa, por lo que la frecuencia de operación del correlador con el código $g'_k[\tau]$ es menor. En consecuencia, para longitudes similares es preferible el empleo de familias con un número K menor de códigos.

xc2vp100	LS(2,16,4) L=79	LS(2,32,4) L=159	LS(2,8,8) L=71	LS(2,16,8) L=143	LS(2,4,16) L=67	LS(2,8,16) L=135
Slices	417	615	538	782	807	1255
LUTs	684	939	858	1171	1274	1797
SRL16	164	300	266	452	466	838
IOBs	36	39	38	41	44	47
freq. max. f_{FPGA}	122.115 MHz	123.244 MHz	86.648 MHz	97.694 MHz	76.144 MHz	76.752 MHz

Tabla 4.13: Resultados de implementación empleando el ELSC propuesto para la detección de códigos $LS(2, L_0, K)$ generados según [SBH01].

Finalmente, al igual que se ha hecho con las propuestas de implementación previas, se compara este correlador con un directo serie. Los resultados de ocupación y frecuencias de operación de ambas implementaciones aparecen en la tabla 4.14 y concuerdan con los teóricos mostrados en la tabla 4.12. Esto es, el ELSC realiza menos operaciones que el correlador directo, lo que se traduce en una frecuencia de trabajo mayor. A cambio, el ELSC necesita más recursos; sin embargo, obsérvese que el porcentaje de ocupación de LUTs y slices es tan sólo del 1 % en la FPGA xc2vp100 seleccionada, por lo que no supone ninguna limitación dicho aumento de recursos.

xc2vp100	LS(2,32,4)	Directo
Slices	615	72
LUTs	939	118
SRL16	300	0
Bloques RAM	0	3
IOBs	39	27
freq. max. f_{FPGA}	123.244 MHz	137.646 MHz
Rendimiento (Mmuestras/s)	123.244	0.865
Tiempo de correlación	8.11 ns	1.15 μs

Tabla 4.14: Comparativa de recursos y tiempos de ejecución empleando un ELSC y correlador directo serie para la detección de códigos LS [SBH01] de longitud $L = 159$.

4.3.2. Algoritmos eficientes asociados a códigos LS generados a partir de CSS

También para códigos LS obtenidos a partir de CSS según [ZYH05] se ha propuesto un esquema de generación (ELSG) y correlación eficiente (ELSC). La figura 4.18 muestra la estructura propuesta para el generador, obtenido según los pasos que se detallan a continuación:

- Paso 1: Generación de los M CSS incorrelados, cada uno con M secuencias, que componen el código LS. Para ello, pueden utilizarse los M generadores eficientes M-ESSG descritos en la sección 4.2.

- Paso 2: Cada ESSG genera simultáneamente M secuencias $s_{i,j}; 0 \leq j \leq M-1$, que son sumadas con diferentes desfases. Como resultado se obtienen M nuevas secuencias, una por cada ESSG, con transformada Z:

$$S'_i[z] = \sum_{j=0}^{M-1} z^{-j(ML_0+W)} S_{i,j}[z];\ 0 \leq i \leq M-1 \tag{4.11}$$

- Paso 3: Un conjunto de M multiplexores controlados por el vector $\pi_n = \{\pi_{n,i}\} = \{(n+i) \bmod M; 0 \leq n, i \leq M-1\}$ determinan los retardos a aplicar a cada una de las secuencias S'_i según:

$$S''_i[z] = z^{-iL_0} \cdot \sum_{j=0}^{M-1} z^{-j(ML_0+W)} S_{\pi_{n,i},j}[z];\ 0 \leq i \leq M-1 \tag{4.12}$$

- Paso 4: Finalmente cada salida del paso 3 es multiplicada por el elemento $h_{m,i}$ correspondiente de la columna h_m de una matriz de Hadamard de tamaño $M \times M$, sumando las secuencias resultantes según (4.13) para obtener el código LS final.

$$G_{n,m}[z] = \sum_{i=0}^{M-1} h_{m,i} \cdot z^{-iL_0} \cdot \left[\sum_{j=0}^{M-1} z^{-j(ML_0+W)} S_{\pi_{n,i},j} \right] \tag{4.13}$$

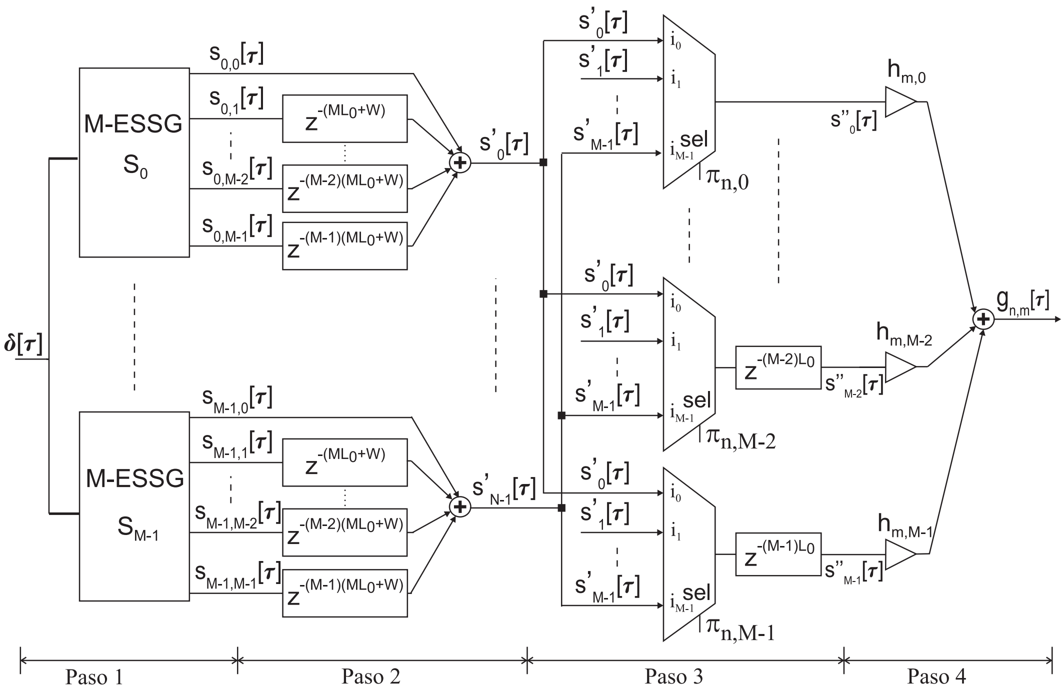


Figura 4.18: Diagrama de bloques de un generador eficiente de códigos LS (ELSG) generados según [ZYH05].

Si se modifica el algoritmo original para que en vez de generar los códigos directos genere las versiones reflejadas $g_{n,m}[L-1-\tau]$ y se aplica la transformada Z sobre el algoritmo resultante se tiene un filtro acoplado a las secuencias directas $g_{n,m}[\tau]$, que puede expresarse como (4.14).

$$C_{R,G_{n,m}}[z] = \sum_{i=0}^{M-1} h_{m,i} \cdot z^{-(M-1-i)L_0} \cdot \left[\sum_{j=0}^{M-1} z^{-(M-1-j)(ML_0+W)} C_{R,S_{\pi_{n,i},j}}[z] \right] \tag{4.14}$$

La implementación de este filtro correlador se representa en la figura 4.19 y los pasos para su obtención son:

- Paso 1: Correlación de la señal de entrada $r[\tau]$ con cada uno de los M CSS incorrelados. Para realizar estas correlaciones de modo eficiente se necesitan M correladores M-ESSCs. Cada M-ESSC realiza la correlación simultánea de la señal de entrada con las

M secuencias de un conjunto S_i, de modo que al final de este paso las correlaciones $\{C_{r,s_{i,j}}[\tau]; 0 \leq i, j \leq M-1\}$ están disponibles.

- Paso 2: Los resultados de correlación anteriores se suman con diferentes retardos, de modo que se obtienen M correlaciones parciales con transformadas Z:

$$C_{R,S'_i}[z] = \sum_{j=0}^{M-1} z^{-(M-1-j)(ML_0+W)} C_{R,S_{i,j}}[z];\ 0 \leq i \leq M-1 \qquad (4.15)$$

- Paso 3: Un conjunto de multiplexores controlado por el vector π_n usado en la generación del código a detectar, determina los retardos $z^{-(M-1-i)L_0}$ a aplicar a cada correlación parcial $C_{R,S'_i}[z]$ según:

$$C_{R,S''_i}[z] = z^{-(M-1-i)L_0} \cdot \left[\sum_{j=0}^{M-1} z^{-(M-1-j)(ML_0+W)} C_{R,S_{\pi_{n,i},j}} \right];\ 0 \leq i \leq M-1 \quad (4.16)$$

- Paso 4: Cada salida del paso anterior se multiplica por el elemento $h_{m,i}$ correspondiente de la matriz de Hadamard, y finalmente se suman según (4.14) para obtener el resultado de correlación con el código $g_{n,m}[\tau]$.

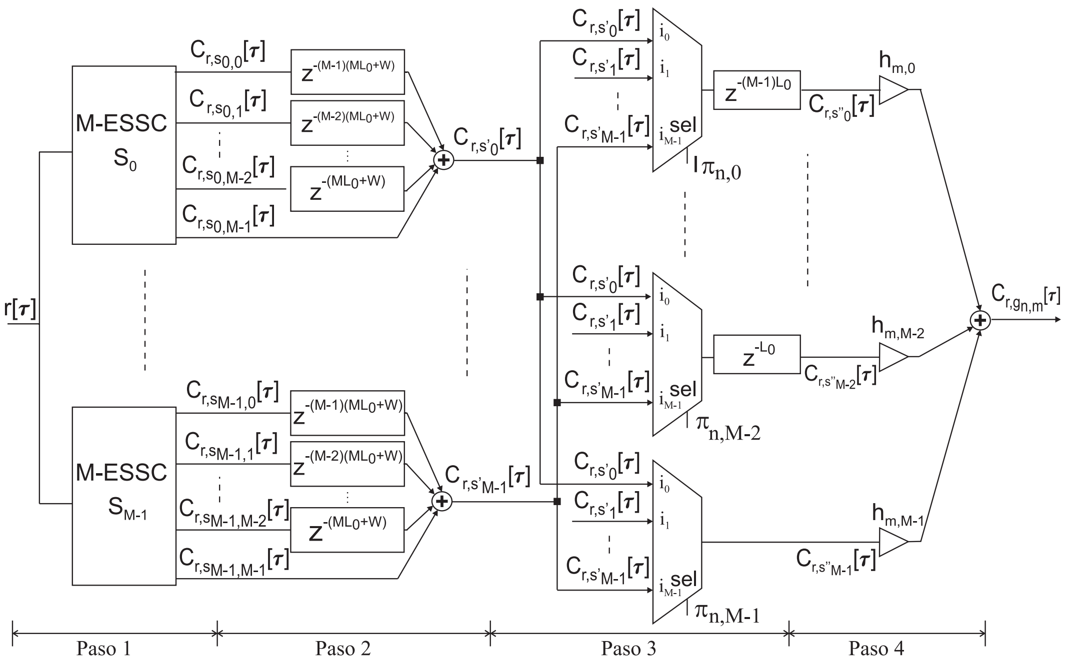


Figura 4.19: Diagrama de bloques de un correlador eficiente de códigos LS (ELSC) generados según [ZYH05].

Una comparación entre el número de multiplicaciones y sumas a realizar en el ELSC propuesto y un correlador directo serie puede encontrarse en la tabla 4.15. Como ya se ha comentado para el ELSC acoplado a códigos LS [SBH01], al tomar los códigos LS valores

$\{-1, 0, 1\}$ las operaciones de multiplicación pueden reducirse a sumas y restas. También en la tabla 4.15 se muestra el número total de posiciones de memoria necesarias para almacenar los datos en ambos correladores. Como ejemplo, en la figura 4.20 se representa gráficamente las necesidades computacionales para una familia LS de $K = 16$ códigos y un ancho de bus de datos de entrada de $DW = 8$ bits. Tanto en dicha figura como en la tabla 4.15 puede observarse la disminución importante del número de operaciones que conlleva el ELSC propuesto frente al correlador directo, aunque por contra, el ELSC requiere más recursos de memoria.

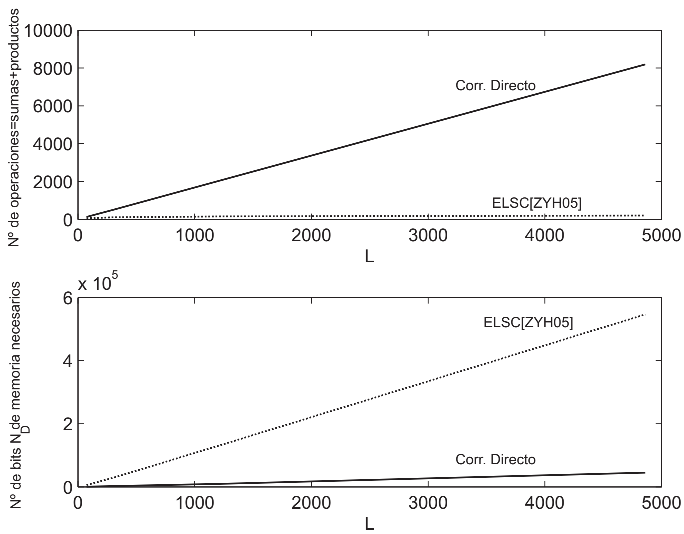


Figura 4.20: Comparativa del número de operaciones y recursos de memoria requeridos por un correlador directo serie y un ELSC [ZYH05], para un total de $K = 16$ usuarios simultáneos con $DW = 8$.

Implementación	Directo serie	ELSC	
Productos	$M^{(N+2)}$	Paso 1	$\frac{N \cdot M^2}{2} log_2(M)$
		Paso 4	M
		TOTAL	$M(1 + \frac{N \cdot M}{2} log_2(M))$
Sumas	$M^{(N+2)} - 1$	Paso 1	$N \cdot M^2 \cdot log_2(M)$
		Paso 2	$M(M-1)$
		Paso 4	$M - 1$
		TOTAL	$N \cdot M^2 \cdot log_2(M) + M^2 - 1$
Bits de memoria	$(N_{SM} \cdot O_f \cdot DW + 2)(M^{N+2} + (M-1)W) +$ $+(DW + Nm + 2m)$	Paso 1	$N_{SM} \cdot O_f \left[(M^{N+1} - M^N)DW + \frac{M^3 - M^2}{2} \left(\frac{M^{N-1}-1}{M-1} DW + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right) \right]$
		Paso 2	$N_{SM} \cdot O_f \cdot (DW + mN) \cdot \left(\frac{(M^{N+1}+W)(M^3-M^2)}{2} \right)$
		Paso 3	$N_{SM} \cdot O_f \cdot (DW + mN + m) \cdot \left(\frac{M^{N+2} - M^{N+1}}{2} \right)$
		TOTAL	$N_{SM} \cdot O_f \left[(M^{N+1} - M^N)DW + \frac{M^3 - M^2}{2} \left(\frac{M^{N-1}-1}{M-1} DW + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right) \right] +$ $+N_{SM} \cdot O_f \left[(DW + mN) \cdot \left(\frac{M^{N+4} - M^{N+3} + M^{N+2} - M^{N+1} + WM^3 - WM^2}{2} \right) + m \left(\frac{M^{N+2} - M^{N+1}}{2} \right) \right]$

Tabla 4.15: Comparativa de las necesidades computacionales de un correlador directo serie y el ELSC propuesto para la correlación de códigos LS de longitud $L = M^2 L_0 + (M-1)W$, generados según [ZYH05].

Resultados de implementación del ELSC propuesto

Teniendo en cuenta las características de este nuevo correlador, los puertos de entrada y salida al mismo son los representados en la figura 4.21. La diferencia frente al correlador propuesto para códigos LS generados a partir de pares Golay [SBH01] reside en que en este nuevo diseño se necesitan M semillas, cada una con $m \cdot N$ bits, correspondientes a los M CSS mutuamente incorrelados que componen el código LS. Estas semillas se proporcionan concatenadas por un único puerto. Los M valores del vector π_n se proporcionan en formato binario con m bits y el tamaño de la columna h_m de la matriz de Hadamard usada en la generación es de M bits. Por otro lado, además de los parámetros genéricos correspondientes al ancho de bus de datos DW, el número de períodos del símbolo de modulación N_{SM}, el factor de sobremuestreo O_f, se puede configurar el número M de CSS incorrelados y el tamaño W de las zonas de ceros a insertar entre cada conjunto de M secuencias $s_{i,j}$. El parámetro M afecta a la longitud final del código $L = M^2 L_0 + (M-1)W$, así como al número $K = M^2$ de códigos ortogonales generalizados disponibles en la familia. El valor de $W \leq L_0 - 1$ determina el tamaño de la zona libre de interferencias alrededor del origen.

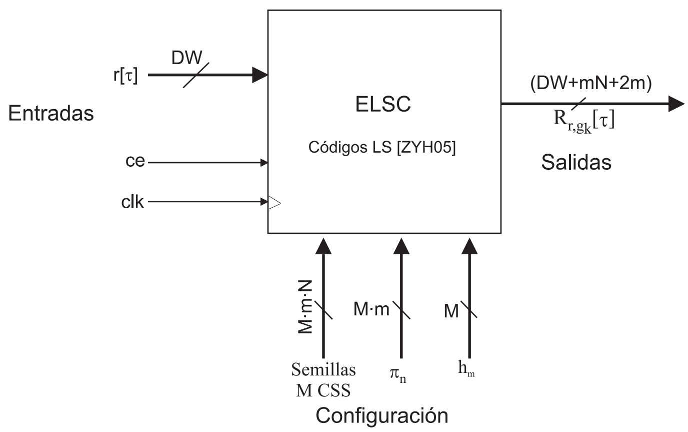


Figura 4.21: Puertos de entrada y salida del ELSC propuesto para códigos LS [ZYH05].

La tabla 4.16 muestra los recursos consumidos por el correlador propuesto para distintas longitudes de código y número de usuarios simultáneos posibles. En todos los casos se ha considerado $DW = 8$, $N_{SM} = O_f = 1$ y $W = L_0 - 1$. Las dos primeras columnas de resultados corresponden a la correlación con códigos LS con $K = 4$ y longitudes $L = 79$ y $L = 159$ bits. Se trata por tanto de los mismos códigos que los evaluados en las columnas segunda y tercera de la tabla 4.13 para el ELSC [SBH01]. Los resultados son similares, algo mejores para la nueva propuesta ELSC [ZYH05] debido a la distinta disposición de los retardos. Las siguientes columnas de la tabla 4.16 han sido calculadas para familias con $K = 16$ códigos LS. De nuevo se observa aquí que, para longitudes de código similar, se necesitan menos recursos y se consiguen mayores frecuencias de operación con aquellas familias que disponen de un número K menor de códigos.

xc2vp100	LS(2,16,4) L=79	LS(2,32,4) L=159	LS(4,4,16) L=73	LS(4,16,16) L=301	LS(4,64,16) L=1213
Slices	381	561	883	2518	8693
LUTs	565	797	1306	3324	9946
SRL16	161	284	492	1740	7464
IOBs	36	39	44	54	64
freq. max. f_{FPGA}	120.525 MHz	114.574 MHz	83.843 MHz	84.203 MHz	85.668 MHz

Tabla 4.16: Resultados de implementación empleando el ELSC propuesto para la detección de códigos $LS(M, L_0, K)$ generados según [ZYH05].

Finalmente, la tabla 4.17 recoge una comparativa entre este correlador y uno directo serie, para la correlación de códigos $LS(4, 16, 16)$ y un ancho de bus de datos de entrada $DW = 8$ bits. De nuevo se comprueba que en cuanto a tiempo de cómputo es más eficiente el correlador ELSC propuesto.

xc2vp100	LS(4,16,16)	Directo
Slices	2518	72
LUTs	3324	120
SRL16	1740	0
Bloques RAM	0	3
IOBs	54	28
freq. max. f_{FPGA}	84.203 MHz	135.031 MHz
Rendimiento (Mmuestras/s)	84.203	0.45
Tiempo de correlación	11.88 ns	2.23 μs

Tabla 4.17: Comparativa de recursos y tiempos de ejecución empleando un ELSC y correlador directo serie para la detección de códigos LS [ZYH05] de longitud $L = 301$.

LS generadas a partir de CSS con $P > 1$

Según se comentaba en el capítulo 2 el algoritmo de generación de códigos LS según [ZYH05] se ha particularizado para el caso de $P = 1$, en donde P indica el número de iteraciones a realizar para obtener mediante concatenación secuencias más largas, que a su vez se concatenan según la ecuación de generación (4.13). En definitiva, utilizar un número

$P > 1$ supone concatenar varias veces la misma secuencia $s_{i,j}$ para obtener un mayor número K de códigos de mayor longitud L. Sin embargo, incrementar P supone aumentar la complejidad del algoritmo de generación, y por ende el de correlación, innecesariamente ya que existen otros mecanismos para conseguir más códigos de mayor longitud en la familia (basta con incrementar el número M de códigos CSS mutuamente incorrelados o la longitud L_0 de dichos CSS). En cualquier caso, a modo de ejemplo la figura (4.22) muestra el correlador asociado a un ELSC obtenido con $P = 2$ a partir de las ecuaciones (4.17) y (4.18).

$$g_n^{(i,j)}[z] = \sum_{k=0}^{M-1} h_{i,k} S_{\pi_{n,k},j}[z] z^{-kML_0} \tag{4.17}$$

$$G_{n,m}[z] = \sum_{i=0}^{M-1} h_{m,i} \cdot z^{-iML_0} \cdot \left[\sum_{j=0}^{M-1} z^{-j(M^2L_0+W)} S_{\pi_{n,i},j} \right] \tag{4.18}$$

Los términos $S_{i,j}[z]$ de la ecuación (4.18) se sustituyen por los términos $g_n^{(i,j)}[z]$ de (4.17) obteniendo de este modo códigos más largos. Además como hay M conjuntos $g_n^{(i,j)}[z]$ el número de códigos de la familia LS se multiplica por M.

4.4. Conclusiones

Este capítulo se ha dedicado al desarrollo y mejora de algoritmos eficientes de correlación asociados a esquemas de codificación derivados de CSS, así como a la implementación hardware de dichos algoritmos en arquitecturas configurables. La ventaja de estos algoritmos respecto a métodos convencionales de correlación es que reducen sobremanera el número de operaciones a realizar sobre las señales recibidas, y por tanto, resultan más adecuados para el proceso de señales ultrasónicas en tiempo real.

Todos los correladores eficientes propuestos se han implementado sobre una FPGA xc2vp100 de Xilinx, detallando en cada caso la cantidad de recursos consumidos y tiempos de ejecución. Además, estos valores se han comparado con los obtenidos en igualdad de condiciones por un correlador directo serie basado en bloques de memoria interna SelectRAM de la FPGA.

Para los CSS se ha realizado una modificación de los algoritmos de generación (ESSG) y correlación (ESSC) eficientes desarrollados en [DMUH+07b] obteniendo arquitecturas regulares, con la misma geometría interna, que derivan en una sencilla implementación genérica en hardware, reduciéndose además el tiempo de diseño frente al requerido por los algoritmos originales. Se han propuesto dos posibles implementaciones, una en la que el número M de secuencias del conjunto y su longitud L se especifican al sintetizar el diseño y que consume sólo los recursos necesarios para esos valores; y otra que permite seleccionar en tiempo real los valores de M y L hasta unos valores máximos.

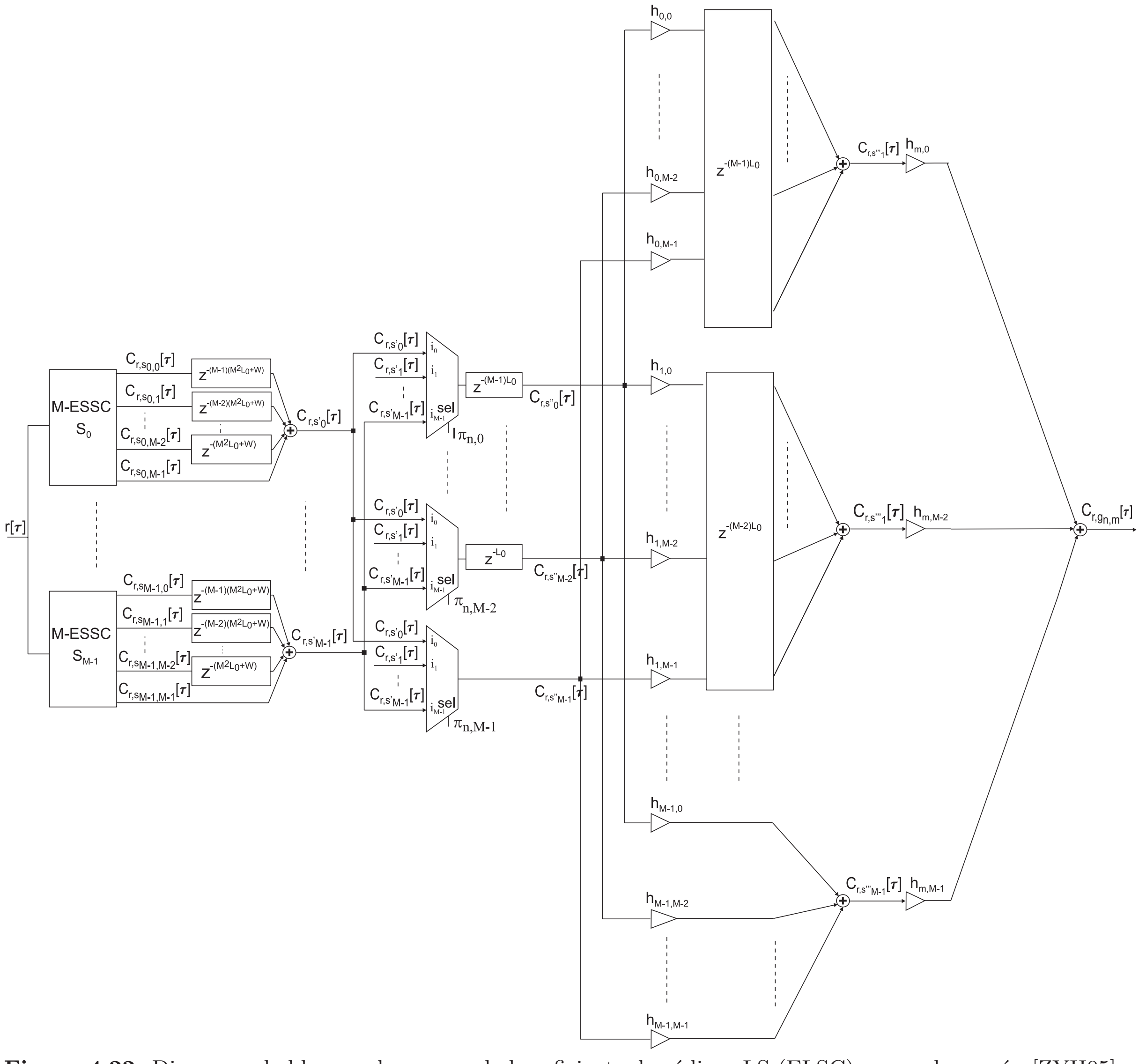


Figura 4.22: Diagrama de bloques de un correlador eficiente de códigos LS (ELSC) generados según [ZYH05] y con $P = 2$.

Por otro lado, el ESSC se ha adaptado al esquema de transmisión mediante macro-secuencias comprobando que, tanto para el caso de concatenación como para el de entrelazado, se consumen menos recursos y se obtienen resultados de correlación en un tiempo menor si se utilizan macro-secuencias con el menor número M posible de secuencias en el conjunto, esto es, con $M = 2$. Obsérvese que también desde el punto de vista de la cota θ de correlación es preferible el uso de macro-secuencias con $M = 2$.

Otra aportación importante de este capítulo es la propuesta de algoritmos de generación y correlación eficientes de códigos LS, ya sean generados a partir de parejas Golay [SBH01] o de CSS [ZYH05]. Los algoritmos propuestos hacen uso de la relación entre estos códigos y los CSS para, a partir del ESSG y ESSC, reducir el número de operaciones a llevar a cabo. La elección de un método u otro ([SBH01] frente a [ZYH05]) no depende tanto de los resultados de implementación de los generadores y correladores desarrollados como de las propiedades intrínsecas de los códigos, evaluadas en el capítulo anterior. Para ambos métodos de generación, las implementaciones propuestas permiten elegir el número de códigos de la familia y la longitud de los mismos en tiempo de síntesis.

Capítulo 5

Nuevo algoritmo de generación y correlación de pares T-ZCZ

En este capítulo se propone un nuevo sistema de codificación con parejas T-ZCZ obtenidas a partir de CSS. Estos códigos se distinguen de otros por poseer tres zonas de correlación cero, una en el origen y dos en las terminaciones de la función de correlación aperiódica, según se indica en la figura 5.1. La ventaja del mecanismo de generación aquí propuesto frente al desarrollado por Chao Zhang et al. en [ZLH04, ZLH05] reside en la utilización de los algoritmos eficientes de generación y correlación asociados a CSS para reducir el número de operaciones a realizar. Además, para determinadas longitudes de códigos, se obtienen zonas libres de interferencias de mayor tamaño que las obtenidas en la propuesta original de Chao Zhang.

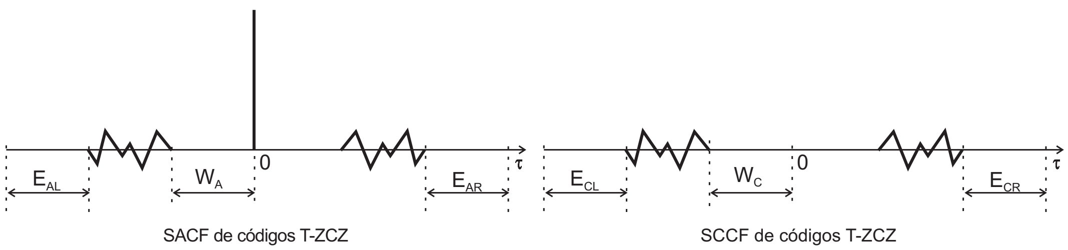


Figura 5.1: SACF y SCCF de códigos T-ZCZ.

La estructura del capítulo es la siguiente: en la primera sección se describen tres métodos distintos de construcción de pares T-ZCZ. En las siguientes secciones se estudian las propiedades de correlación de los pares obtenidos con cada método y se proponen para cada caso algoritmos eficientes de generación y correlación. En cuanto a las propiedades de correlación, los aspectos a evaluar son el tamaño de las zonas libres de interferencias, la magnitud y número de valores no nulos en la zona con interferencias y la viabilidad de estos códigos cuando se pierde el inicio de los mismos debido, por ejemplo, a la presencia de un obstáculo en la zona ciega de un sistema de eco-detección. Por otro lado, cada correlador propuesto es comparado con uno directo convencional considerando tanto el número de operaciones a realizar como la cantidad de memoria requerida para llevar a cabo la correlación.

5.1. Nuevo mecanismo de generación de pares T-ZCZ

En [ZLH04] los pares T-ZCZ se obtienen a partir de una matriz de generación Δ. En la práctica el mecanismo de construcción de esta matriz es el mismo que el propuesto por Tseng y Liu [TL72] para la obtención de M CSS mutuamente incorrelados (UCSS). Sabiendo esto, puede obtenerse una nueva matriz Δ mediante la utilización de CSS generados según el método propuesto en [DMUH$^+$07b] (véase la sección B.4). La ventaja de utilizar estos últimos CSS es la disponibilidad de generadores y correladores eficientes (ESSG y ESSC respectivamente). Por otro lado, para la construcción de la nueva matriz Δ pueden usarse dos alternativas: M UCSS o uno solo de mayor longitud. La diferencia reside principalmente en el número y forma de combinar los ESSGs y ESSCs asociados para la generación y ulterior correlación de los pares T-ZCZ. Las dos vías mencionadas son:

- *Un único CSS:* cada fila de la matriz de generación $\Delta^{(M,L_0}$ de tamaño $(M \times L_0)$ es una secuencia del CSS S_i, en donde $M = 2^m$ es el número de secuencias del conjunto y $L_0 = M^N$ la longitud de las secuencias complementarias, con $m, N \in \mathbb{N} - \{0\}$. Además, considerando todos los posibles CSS de una misma longitud y número M de secuencias, estén o no incorrelados, hay L_0 posibilidades $(\{S_i = s_{i,j}[l]; 0 \leq i, l \leq L_0; 0 \leq j \leq M-1\})$, por lo que pueden conseguirse L_0 matrices de generación distintas de una misma longitud.

$$\Delta^{(M,L_0} = S_i^{(M,L_0} = \begin{pmatrix} s_{i,0}[\tau] \\ s_{i,1}[\tau] \\ \vdots \\ s_{i,M-1}[\tau] \end{pmatrix} = \begin{pmatrix} s_{i,0}[0] & s_{i,0}[1] & \cdots & s_{i,0}[L_0-1] \\ s_{i,1}[0] & s_{i,1}[1] & \cdots & s_{i,1}[L_0-1] \\ \vdots & \vdots & \ddots & \vdots \\ s_{i,M-1}[0] & s_{i,M-1}[1] & \cdots & s_{i,M-1}[L_0-1] \end{pmatrix} \tag{5.1}$$

- *M UCSS:* puede obtenerse una nueva matriz $\Delta^{(M,M\cdot L_0}$ de tamaño $(M \times M \cdot L_0)$ concatenando M conjuntos incorrelados $\{S_i; 0 \leq i \leq M-1\}$[1], de modo que las L_0 primeras columnas corresponden a los bits del conjunto S_0, las L_0 siguientes a las del conjunto S_1 y así sucesivamente. El valor del subíndice i que indica el orden con el que se concatenan los CSS se calcula a partir de la representación decimal

[1]Cuando se trabaja con conjuntos incorrelados entre sí, es habitual utilizar el subíndice i para indicar el i-ésimo conjunto incorrelado [SBH01, ZYH05]. Una representación más correcta sería $\{S_{(p+i\frac{L_0}{M}) \bmod L_0}; 0 \leq i \leq M-1\}$ siendo $\{S_p; 0 \leq p \leq L_0-1\}$ un CSS cualquiera.

de los M primeros bits $\{w_{1,1}, w_{2,1}, \cdots, w_{m,1}\}$ de la semilla con la que se generaron, siendo $w_{1,1}$ el bit más significativo y traduciendo como 0 los valores -1. Podrían conseguirse $\frac{L_0}{M}$ matrices $\Delta^{(M,M\cdot L_0}$ distintas variando los valores de los coeficientes restantes $\{w_{1,2}, w_{2,2}, \cdots, w_{m,2}, \cdots, w_{1,N}, w_{2,N}, \cdots, w_{m,N}\}$.

$$\Delta^{(M,M\cdot L_0} = (S_0 \mid S_1 \mid \cdots \mid S_{M-1}) = \begin{pmatrix} s_{0,0} & s_{1,0} & \cdots & s_{M-1,0} \\ s_{0,1} & s_{1,1} & \cdots & s_{M-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{0,M-1} & s_{1,M-1} & \cdots & s_{M-1,M-1} \end{pmatrix} =$$
$$\begin{pmatrix} s_{0,0}[0] & \cdots & s_{0,0}[L_0-1] & s_{1,0}[0] & \cdots & s_{1,0}[L_0-1] & \cdots & s_{M-1,0}[0] & \cdots & s_{M-1,0}[L_0-1] \\ s_{0,1}[0] & \cdots & s_{0,1}[L_0-1] & s_{1,1}[0] & \cdots & s_{1,1}[L_0-1] & \cdots & s_{M-1,1}[0] & \cdots & s_{M-1,1}[L_0-1] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ s_{0,M-1}[0] & \cdots & s_{0,M-1}[L_0-1] & s_{1,M-1}[0] & \cdots & s_{1,M-1}[L_0-1] & \cdots & s_{M-1,M-1}[0] & \cdots & s_{M-1,M-1}[L_0-1] \end{pmatrix} \tag{5.2}$$

En función de cómo se agrupen los bits de las matrices de generación $\Delta^{(M,L_0}$ ó $\Delta^{(M,M\cdot L_0}$ se obtienen tres métodos distintos de construcción de pares T-ZCZ. Para mayor claridad se ha añadido un sub-índice que indica el número de método (1, 2 ó 3) utilizado en la generación de los códigos T-ZCZ, y una tilde sobre el sub-índice cuando se hayan obtenido a partir de la matriz $\Delta^{(M,M\cdot L_0}$ (la ausencia de tilde corresponde a pares T-ZCZ construidos usando la matriz $\Delta^{(M,L_0}$). Por ejemplo, los pares T-ZCZ$_{1'}$ han sido obtenidos mediante el método 1 y utilizando M UCSS, y los pares T-ZCZ$_1$ también mediante el método 1, pero a partir de un único CSS. Cuando no sea necesario hacer distinción del método empleado se hablará de códigos T-ZCZ en general, sin emplear sub-índices. Del mismo modo, se hablará de una matriz Δ de generación genérica cuando no se requiera especificar la composición interna de dicha matriz, esto es, cuando las transformaciones a aplicar sean válidas tanto para $\Delta^{(M,L_0}$ como para $\Delta^{(M,M\cdot L_0}$.

Método 1. Se reescribe la matriz Δ como:

$$\Delta = \begin{pmatrix} A_0 \\ A_1 \\ \vdots \\ A_{M-1} \end{pmatrix} \tag{5.3}$$

En donde A_i indica la fila i-ésima de la matriz Δ. El conjunto de códigos T-ZCZ está formado por los pares $E_0 = (A_0, A_1)$, $E_1 = (A_2, A_3)$, $\cdots$, $E_{M/2-1} = (A_{M-2}, A_{M-1})$, y los pares que son compañeros a los anteriores $E_{M/2} = (\overline{A_1}, \overline{-A_0})$, $E_{M/2+1} = (\overline{A_3}, \overline{-A_2})$, $\cdots$, $E_{M-1} = (\overline{A_{M-1}}, \overline{-A_{M-2}})$. La longitud L de estos pares coincide con la longitud de cada fila A_i de la matriz Δ y el número de pares con ZCZ en las sumas de las funciones de correlación es el mismo que el número M de secuencias complementarias del conjunto. Particularizando la matriz Δ en las matrices $\Delta^{(M,L_0}$ y $\Delta^{(M,M\cdot L_0}$ se tienen los códigos T-ZCZ$_1$ y T-ZCZ$_{1'}$ respectivamente:

- T-ZCZ$_1$ ($\Delta = \Delta^{(M,L_0)}$): Los primeros $\frac{M}{2}$ pares de una familia T-ZCZ$_1$ $\{E_k = (e_{k,0}, e_{k,1}); 0 \leq k \leq M-1\}$ obtenida mediante el método 1 y a partir de un único CSS $S_i = \{s_{i,0}, s_{i,1}, \cdots, s_{i,M-1}\}$, se componen de las secuencias: $E_0 = (e_{0,0}, e_{0,1}) = (s_{i,0}, s_{i,1})$, $E_1 = (e_{1,0}, e_{1,1}) = (s_{i,2}, s_{i,3})$, $\cdots$, $E_{M/2-1} = (e_{M/2-1,0}, e_{M/2-1,1}) = (s_{i,M-2}, s_{i,M-1})$. Los siguientes $\frac{M}{2}$ pares se obtienen de las secuencias de un conjunto $S_{i'} = \{\overline{s_{i,1}}, -\overline{s_{i,0}}, \overline{s_{i,2}}, -\overline{s_{i,1}}, \cdots, \overline{s_{i,M-1}}, -\overline{s_{i,M-2}}\}$ incorrelado con S_i, de modo que $E_{M/2} = (e_{M/2,0}, e_{M/2,1}) = (\overline{s_{i,1}}, -\overline{s_{i,0}})$, $\cdots$, $E_{M-1} = (e_{M-1,0}, e_{M-1,1}) = (\overline{s_{i,M-1}}, -\overline{s_{i,M-2}})$. En (5.4) se resume el proceso de construcción de los códigos T-ZCZ$_1$. Obsérvese que la longitud de los pares obtenidos coincide con la de las secuencias del CSS, esto es, $L = L_0$.

$$\begin{cases} e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2} = s_{i,j}, & 0 \leq j \leq M-1 \\ e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2} = (-1)^{j+1} \cdot \overline{s_{i,j}}, & 0 \leq j \leq M-1 \end{cases} \tag{5.4}$$

- T-ZCZ$_{1'}$ ($\Delta = \Delta^{(M,M \cdot L_0)}$): Se obtienen en este caso los pares T-ZCZ$_{1'}$ indicados en (5.5), cuya longitud es $L = M \cdot L_0$.

$$\begin{cases} e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2} = [s_{0,j} \mid s_{1,j} \mid \cdots \mid s_{M-1,j}], & 0 \leq j \leq M-1 \\ e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2} = (-1)^{j+1} \cdot [\overline{s_{M-1,j}} \mid \overline{s_{M-2,j}} \mid \cdots \mid \overline{s_{0,j}}], & 0 \leq j \leq M-1 \end{cases} \tag{5.5}$$

Método 2. Se divide la matriz Δ en dos sub-matrices Δ_L y Δ_R de idéntico tamaño, siendo $\Delta = [\Delta_L \mid \Delta_R]$ -véase (5.6)-. Sabiendo esto, los pares $E_0 = (A_{0_L}, A_{0_R})$, $E_1 = (A_{1_L}, A_{1_R})$, $\cdots$, $E_{M-1} = (A_{M-1_L}, A_{M-1_R})$ constituyen un conjunto T-ZCZ. El número de pares T-ZCZ obtenido es M y su longitud es la mitad de la longitud de las filas de Δ. Específicamente, los pares T-ZCZ$_2$ y T-ZCZ$_{2'}$ son:

$$\Delta = \begin{pmatrix} A_{0_L} & A_{0_R} \\ A_{1_L} & A_{1_R} \\ \vdots & \vdots \\ A_{M-1_L} & A_{M-1R} \end{pmatrix} \tag{5.6}$$

- T-ZCZ$_2$ ($\Delta = \Delta^{(M,L_0)}$): En este caso la secuencia 0 del par T-ZCZ$_2$ j-ésimo corresponde a la mitad derecha de la secuencia $s_{i,j}$ y la secuencia 1 a la mitad izquierda, es decir, $e_{j,0} = ([s_{i,j}[0], s_{i,j}[1], \cdots, s_{i,j}[\frac{L_0}{2} - 1])$ y $e_{j,1} = (s_{i,j}[\frac{L_0}{2}], s_{i,j}[\frac{L_0}{2} + 1], \cdots, s_{i,j}[L_0 - 1])$, donde $0 \leq j \leq M-1$ y $L = \frac{L_0}{2}$.

- T-ZCZ$_{2'}$ ($\Delta = \Delta^{(M,M \cdot L_0)}$): Pueden obtenerse $0 \leq j \leq M-1$ pares T-ZCZ$_{2'}$ de longitud $L = \frac{M}{2} L_0$ como $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j} | s_{1,j} | \cdots | s_{\frac{M}{2}-1,j}], [s_{\frac{M}{2},j} | s_{\frac{M}{2}+1,j} | \cdots | s_{M-1,j}])$.

Método 3. Finalmente, pueden conseguirse M pares T-ZCZ dividiendo nuevamente en dos cada una de las sub-matrices Δ_L y Δ_R, de modo que $\Delta_L = [\Delta_{0_L} \mid \Delta_{1_L}]$ y $\Delta_R = [\Delta_{0_R} \mid \Delta_{1_R}]$ según se indica en (5.7). Los pares $E_0 = (A_{0,0_L}, A_{0,1_L})$, $E_1 = (A_{1,0_L}, A_{1,1_L})$, $\cdots$, $E_{M-1} = (A_{M-1,0_L}, A_{M-1,1_L})$ forman un conjunto T-ZCZ; y los pares $(A_{0,0_R}, A_{0,1_R})$, $(A_{1,0_R}, A_{1,1_R})$, $\cdots$, $(A_{M-1,0_R}, A_{M-1,1_R})$ otro conjunto T-ZCZ distinto. Sabiendo esto, los pares T-ZCZ$_3$ y T-ZCZ$_{3'}$ son:

$$\Delta_L = \begin{pmatrix} A_{0,0_L} & A_{0,1_L} \\ A_{1,0_L} & A_{1,1_L} \\ \vdots & \vdots \\ A_{M-1,0_L} & A_{M-1,1_L} \end{pmatrix}; \quad \Delta_R = \begin{pmatrix} A_{0,0_R} & A_{0,1_R} \\ A_{1,0_R} & A_{1,1_R} \\ \vdots & \vdots \\ A_{M-1,0_R} & A_{M-1,1_R} \end{pmatrix} \tag{5.7}$$

- T-ZCZ$_3$ ($\Delta = \Delta^{(M,L_0)}$): Utilizando la mitad izquierda $\Delta_L^{(M,L_0}$ de la matriz $\Delta^{(M,L_0}$ puede construirse una familia de códigos T-ZCZ$_3$ de modo que $e_{j,0} = ([s_{i,j}[0], s_{i,j}[1], \cdots s_{i,j}[\frac{L_0}{4} - 1])$ y $e_{j,1} = (s_{i,j}[\frac{L_0}{4}], s_{i,j}[\frac{L_0}{4} + 1], \cdots s_{i,j}[\frac{L_0}{2} - 1])$. Asimismo, con la mitad derecha $\Delta_L^{(M,L_0}$ puede construirse una familia T-ZCZ$_3$ distinta, en donde $e_{j,0} = ([s_{i,j}[\frac{L_0}{2}], s_{i,j}[\frac{L_0}{2}], \cdots s_{i,j}[\frac{3L_0}{4} - 1])$ y $e_{j,1} = (s_{i,j}[\frac{3L_0}{4}], s_{i,j}[\frac{3L_0}{4} + 1], \cdots s_{i,j}[L_0 - 1])$. La longitud de estos códigos es $L = \frac{L_0}{4}$
- T-ZCZ$_{3'}$ ($\Delta = \Delta^{(M,M \cdot L_0)}$): Se obtienen códigos de longitud $L = \frac{ML_0}{4}$, ya sea a partir de $\Delta_L^{(M,\frac{M}{2} \cdot L_0}$, en cuyo caso los pares T-ZCZ$'_3$ son $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}| \cdots |s_{\frac{M}{4}-1,j}], [s_{\frac{M}{4},j}|s_{\frac{M}{4}+1,j}| \cdots |s_{\frac{M}{2}-1,j}])$, con $0 \leq j \leq M-1$ y $M > 2$; o a partir de $\Delta_R^{(M,\frac{M}{2} \cdot L_0}$, obteniendo otra familia $E_j = (e_{j,0}, e_{j,1}) = ([s_{\frac{M}{2},j}|s_{\frac{M}{2}+1,j}| \cdots |s_{\frac{3M}{4}-1,j}], [s_{\frac{3M}{4},j}|s_{\frac{3M}{4}+1,j}| \cdots |s_{M-1,j}])$ distinta de códigos T-ZCZ$'_3$.

Teniendo en cuenta la nomenclatura mostrada en la figura 5.1, las tablas 5.1 a 5.6 ilustran ejemplos de códigos T-ZCZ(L, M, E_{AL}, W_A) generados con los métodos descritos; en donde L es la longitud de las parejas evaluadas, M el número de parejas ortogonales generalizadas disponibles de esa longitud, E_{AL} el tamaño de la IFW lateral derecha en la SACF (en estos códigos $E_{AL} = E_{AR} = E_{CL} = E_{CR}$) y W_A el tamaño de la IFW alrededor del origen (siendo $W_A = W_C$). Nótese que no siempre coincide el tamaño de las IFWs laterales con el de la ventana libre de interferencias alrededor del origen.

En las siguientes secciones se determina el tamaño de las áreas libres de interferencias y se analizan las propiedades de correlación de los pares T-ZCZ propuestos. Para cada método de construcción de pares T-ZCZ hay que considerar el caso de haber utilizado la matriz de generación $\Delta^{(M,L_0}$ ó $\Delta^{(M,M \cdot L_0}$. Sin embargo, con objeto de abreviar, se analizan en mayor profundidad los resultados de emplear la matriz de generación que redunde en una implementación hardware del correlador asociado más eficiente; haciendo un breve resumen al final de cada sección de los resultados y motivos por los que la otra construcción se ha descartado.

$e_{0,0}$	1	1	-1	-1	1	1	1	1	-1	1	1	-1	-1	1	-1	1
$e_{0,1}$	1	1	1	1	1	1	-1	-1	-1	1	-1	1	-1	1	1	-1
$e_{1,0}$	1	-1	-1	1	1	-1	1	-1	-1	-1	1	1	-1	-1	-1	-1
$e_{1,1}$	1	-1	1	-1	1	-1	-1	1	-1	-1	-1	-1	-1	-1	1	1
$e_{2,0}$	-1	1	1	-1	1	-1	1	-1	-1	-1	1	1	1	1	1	1
$e_{2,1}$	-1	1	-1	1	1	-1	-1	1	-1	-1	-1	-1	1	1	-1	-1
$e_{3,0}$	1	1	-1	-1	-1	-1	-1	-1	1	-1	-1	1	-1	1	-1	1
$e_{3,1}$	1	1	1	1	-1	-1	1	1	1	-1	1	-1	-1	1	1	-1

Tabla 5.1: T-ZCZ$_1(16, 4, 6, 6)$ obtenido a partir de un CSS con semilla de generación $w_{1,1} = w_{2,1} = w_{1,2} = w_{2,2} = -1$ y $L_0 = 16$.

$e_{0,0}$	1	1	-1	-1	1	-1	-1	1	1	1	1	1	1	-1	1	-1
$e_{0,1}$	1	1	1	1	1	-1	1	-1	1	1	-1	-1	1	-1	-1	1
$e_{1,0}$	1	-1	-1	1	1	1	-1	-1	1	-1	1	-1	1	1	1	1
$e_{1,1}$	1	-1	1	-1	1	1	1	1	1	-1	-1	1	1	1	-1	-1
$e_{2,0}$	1	-1	-1	1	-1	-1	1	1	-1	1	-1	1	1	1	1	1
$e_{2,1}$	1	-1	1	-1	-1	-1	-1	-1	-1	1	1	-1	1	1	-1	-1
$e_{3,0}$	-1	-1	1	1	1	-1	-1	1	1	1	1	1	-1	1	-1	1
$e_{3,1}$	-1	-1	-1	-1	1	-1	1	-1	1	1	-1	-1	-1	1	1	-1

Tabla 5.2: T-ZCZ$_{1'}(16, 4, 10, 2)$, en donde $M = 4$ UCSS S_i con $L_0 = 4$ se han obtenido a partir de todas las combinaciones posibles de $w_{1,1}$ y $w_{2,1}$.

$e_{0,0}$	1	1	-1	-1	1	1	1	1
$e_{0,1}$	-1	1	1	-1	-1	1	-1	1
$e_{1,0}$	1	1	1	1	1	1	-1	-1
$e_{1,1}$	-1	1	-1	1	-1	1	1	-1
$e_{2,0}$	1	-1	-1	1	1	-1	1	-1
$e_{2,1}$	-1	-1	1	1	-1	-1	-1	-1
$e_{3,0}$	1	-1	1	-1	1	-1	-1	1
$e_{3,1}$	-1	-1	-1	-1	-1	-1	1	1

Tabla 5.3: T-ZCZ$_2(8, 4, 1, 1)$, obtenido a partir de un CSS con semilla de generación $w_{1,1} = w_{2,1} = w_{1,2} = w_{2,2} = -1$ y $L_0 = 16$.

$e_{0,0}$	1	1	-1	-1	1	-1	-1	1
$e_{0,1}$	1	1	1	1	1	-1	1	-1
$e_{1,0}$	1	1	1	1	1	-1	1	-1
$e_{1,1}$	1	1	-1	-1	1	-1	-1	1
$e_{2,0}$	1	-1	-1	1	1	1	-1	-1
$e_{2,1}$	1	-1	1	-1	1	1	1	1
$e_{3,0}$	1	-1	1	-1	1	1	1	1
$e_{3,1}$	1	-1	-1	1	1	1	-1	-1

Tabla 5.4: T-ZCZ$_{2'}(8, 4, 2, 2)$, en donde $M = 4$ UCSS S_i con $L_0 = 4$ se han obtenido a partir de todas las combinaciones posibles de $w_{1,1}$ y $w_{2,1}$.

$e_{0,0}$	1	1	-1	-1	1	1	1	1	-1	1	1	-1	-1	1	-1	1
$e_{0,1}$	1	1	-1	-1	1	1	1	1	1	-1	-1	1	1	-1	1	-1
$e_{1,0}$	1	1	1	1	1	1	-1	-1	-1	1	-1	1	-1	1	1	-1
$e_{1,1}$	1	1	1	1	1	1	-1	-1	1	-1	1	-1	1	-1	-1	1
$e_{2,0}$	1	-1	-1	1	1	-1	1	-1	-1	-1	1	1	-1	-1	-1	-1
$e_{2,1}$	1	-1	-1	1	1	-1	1	-1	1	1	-1	-1	1	1	1	1
$e_{3,0}$	1	-1	1	-1	1	-1	-1	1	-1	-1	-1	-1	-1	-1	1	1
$e_{3,1}$	1	-1	1	-1	1	-1	-1	1	1	1	1	1	1	1	-1	-1

Tabla 5.5: T-ZCZ$_3(16, 4, 9, 1)$ obtenido a partir de un CSS con semilla de generación $w_{1,1} = w_{2,1} = w_{1,2} = w_{2,2} = w_{1,3} = w_{2,3} = -1$ y $L_0 = 64$.

$e_{0,0}$	1	1	-1	-1	1	1	1	1	-1	1	1	-1	-1	1	-1	1
$e_{0,1}$	1	1	-1	-1	-1	-1	-1	-1	-1	1	1	-1	1	-1	1	-1
$e_{1,0}$	1	1	1	1	1	1	-1	-1	-1	1	-1	1	-1	1	1	-1
$e_{1,1}$	1	1	1	1	-1	-1	1	1	-1	1	-1	1	1	-1	-1	1
$e_{2,0}$	1	-1	-1	1	1	-1	1	-1	-1	-1	1	1	-1	-1	-1	-1
$e_{2,1}$	1	-1	-1	1	-1	1	-1	1	-1	-1	1	1	1	1	1	1
$e_{3,0}$	1	-1	1	-1	1	-1	-1	1	-1	-1	-1	-1	-1	-1	1	1
$e_{3,1}$	1	-1	1	-1	-1	1	1	-1	-1	-1	-1	-1	1	1	-1	-1

Tabla 5.6: T-ZCZ$_{3'}(16, 4, 6, 6)$, en donde $M = 4$ UCSS S_i con $L_0 = 16$ se han obtenido fijando los coeficientes $w_{1,2} = w_{2,2} = -1$ y realizando todas las combinaciones posibles de $w_{1,1}$ y $w_{2,1}$.

5.2. Caracterización del método 1 usando $\Delta^{(M,L_0}$

5.2.1. Estudio de las propiedades de correlación

El estudio de las propiedades de correlación de cada método propuesto se ha realizado siguiendo la misma metodología de análisis que la llevada a cabo en el capítulo 3, apartado 3.5, para códigos T-ZCZ generados según [ZLH04], de modo que puedan compararse fácilmente los resultados de ambas propuestas. Así, en las tablas 5.7 a 5.9, y para códigos T-ZCZ$_1$, se muestran las $\mu \leq M$ combinaciones de menor cota de correlación $\theta = \max(\theta_{AC}, \theta_{CC})$ en la zona con interferencias (IW) en función del número M de códigos de la familia y de la longitud L de los mismos. Nótese que el código 0 corresponde a $E_0 = (s_{i,0}, s_{i,1})$ y el código $M-1$ a $E_{M-1} = (\overline{s_{i,M-1}}, \overline{-s_{i,M-2}})$. No se incluye en las tablas la semilla de generación $Wd_N^{(m}$ del CSS que compone la base de los pares T-ZCZ$_1$, ya que no influye en el número de interferencias ni en la cota de correlación de dichos pares, por lo que puede elegirse cualquiera.

Entre los distintos parámetros reflejados en las tablas siguientes, se encuentran las longitudes de las IFWs en la SACF, que coinciden con las obtenidas en la SCCF cuando $\mu = M$. Como las funciones de correlación son simétricas, para simplificar se ha utilizado un único término, E_A en la SACF y E_C en la SCCF, para denotar el tamaño de las zonas de correlación cero laterales derecha e izquierda (es decir, $E_{AL} = E_{AR} = E_A$ y $E_{CL} = E_{CR} = E_C$). Por otro lado, se ha observado que en el área con interferencias (IW) de los pares T-ZCZ$_1$ también aparecen valores nulos; en este sentido, se muestra el porcentaje de interferencias en la IW de la SACF ($\%\frac{Int.IW_{SACF}}{IW}$) y en la IW de la SCCF ($\%\frac{Int.IW_{SCCF}}{IW}$). Obsérvese que para un número de usuarios $\mu = 2$ no hay interferencias en la SCCF, y por tanto $\theta_{CC} = 0$.

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
	2	0.25	0.25	0	0, 3				0 %
16	3	0.5	0.25	0.5	0, 1, 2	6	6	66.67 %	100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
	2	0.25	0.25	0	0, 3				0 %
64	3	0.25	0.25	0.25	0, 1, 2	38	6	31.58 %	42.11 %
	4	0.25	0.25	0.25	0, 1, 2, 3				42.11 %
	2	0.1875	0.1875	0	0, 2				0 %
256	3	0.375	0.1875	0.375	0, 1, 2	102	102	23.53 %	31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %
	2	0.125	0.125	0	0, 2				0 %
1024	3	0.1875	0.125	0.1875	0, 1, 2	614	102	11.73 %	15.64 %
	4	0.1875	0.125	0.1875	0, 1, 2, 3				15.64 %
	2	0.1094	0.1094	0	0, 2				0 %
4096	3	0.2188	0.1094	0.2188	0, 1, 2	1638	1638	8.79 %	11.72 %

Continuación de la Tabla 5.7

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %
16384	2	0.0625	0.0625	0	0, 2	9830	1638	4.39 %	0 %
	3	0.1094	0.0625	0.1094	0, 1, 2				5.86 %
	4	0.1094	0.0625	0.1094	0, 1, 2, 3				5.86 %

Tabla 5.7: Estudio de la zona con interferencias de familias T-ZCZ$_1$ con $M = 4$ códigos.

M = 8

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
64	2	0.25	0.25	0	0, 4	12	12	30.77 %	0 %
	3	0.25	0.25	0.25	0, 3, 4				35.90 %
	4	0.25	0.25	0.25	0, 3, 4, 7				38.46 %
	5	0.5	0.25	0.5	0, 1, 2, 3, 4				38.46 %
	6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5				38.46 %
	7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6				38.46 %
	8	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7				38.46 %
512	2	0.25	0.25	0	1, 5	268	12	26.84 %	0 %
	3	0.3125	0.3125	0.1406	1, 4, 7				38.10 %
	4	0.3125	0.3125	0.1406	0, 3, 4, 7				38.10 %
	5	0.3125	0.3125	0.25	0, 1, 2, 4, 5				38.10 %
	6	0.3125	0.3125	0.25	0, 1, 2, 4, 5, 6				38.10 %
	7	0.3125	0.3125	0.3125	0, 1, 2, 3, 5, 6, 7				38.10 %
	8	0.3125	0.3125	0.3125	0, 1, 2, 3, 4, 5, 6, 7				38.10 %
4096	2	0.1875	0.1875	0	0, 4	780	780	9.94 %	0 %
	3	0.1875	0.18735	0.1523	0, 3, 4				13.89 %
	4	0.1875	0.1875	0.1523	0, 3, 4, 7				13.89 %
	5	0.375	0.1875	0.375	0, 1, 2, 3, 4				13.89 %
	6	0.375	0.1875	0.375	0, 1, 2, 4, 5, 6				13.89 %
	7	0.375	0.1875	0.375	0, 1, 2, 3, 4, 5, 6				13.89 %
	8	0.375	0.1875	0.375	0, 1, 2, 3, 4, 5, 6, 7				13.89 %
32768	2	0.125	0.125	0	1, 5	17164	780	10.87 %	0 %
	3	0.1406	0.1406	0.0732	0, 3, 5				16.62 %
	4	0.1406	0.1406	0.0732	0, 1, 4, 7				16.62 %
	5	0.1875	0.1406	0.1875	0, 1, 4, 5, 7				16.62 %
	6	0.1875	0.1406	0.1875	0, 1, 4, 5, 6, 7				16.62 %
	7	0.1875	0.1406	0.1875	0, 1, 2, 3, 4, 5, 6				16.62 %
	8	0.1875	0.1406	0.1875	0, 1, 2, 3, 4, 5, 6, 7				16.62 %

Tabla 5.8: Estudio de la zona con interferencias de familias T-ZCZ$_1$ con $M = 8$ códigos.

M = 16

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
256	2	0.25	0.25	0	1, 8	24	24	26.09 %	0 %
	3	0.25	0.25	0.125	0, 7, 8				28.99 %
	4	0.25	0.25	0.125	0, 7, 8, 15				28.99 %
	5	0.25	0.25	0.25	0, 3, 5, 6, 8				33.82 %
	6	0.25	0.25	0.25	0, 3, 5, 6, 8, 11				33.82 %
	7	0.25	0.25	0.25	0, 3, 5, 6, 8, 11, 13				33.82 %
	8	0.25	0.25	0.25	0, 3, 5, 6, 8, 11, 13, 14				33.82 %
	10	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				33.82 %
	14	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				33.82 %
	16	0.5	0.25	0.5	0, 1, 2, $\cdots$, 15				33.82 %
4096	2	0.25	0.25	0	3, 11	2072	24	27.11 %	0 %
	3	0.2969	0.25	0.2969	3, 7, 11				36.52 %
	4	0.2969	0.25	0.2969	3, 7, 11, 15				36.52 %
	5	0.3125	0.3125	0.1523	1, 2, 7, 9, 10				40.42 %
	6	0.3125	0.3125	0.1523	1, 2, 7, 9, 10, 15				40.42 %
	7	0.3281	0.3281	0.1523	0, 3, 5, 6, 8, 11, 14				40.42 %
	8	0.3281	0.3281	0.1523	0, 3, 5, 6, 9, 10, 12, 15				40.42 %
	10	0.3281	0.3281	0.25	0, 1, 2, 4, 5, 8, 9, 10, 12, 13				41.62 %
	14	0.3281	0.3281	0.3125	0, 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 15				41.62 %
	16	0.3281	0.3281	0.3281	0, 1, 2, $\cdots$, 15				41.62 %

Tabla 5.9: Estudio de la zona con interferencias de familias T-ZCZ$_1$ con $M = 16$ códigos.

En las tablas anteriores puede observarse que, cuando el número de etapas N del generador es par, el tamaño de las zonas libres de interferencias en los laterales es el mismo que el tamaño de las zonas sin interferencias alrededor del origen, esto es, $E_A = E_C = W_A = W_C$. Sin embargo, cuando el número de etapas N es impar el tamaño de dichas zonas no coincide ($E_A = E_C \neq W_A = W_C$). La ecuaciones (5.8) y (5.9) especifican los valores de estas zonas sin interferencias en función de M y N.

- Cuando N es par

$$W_A = W_C = E_A = E_C = \left[\frac{\frac{M}{2}(M^N - 1)}{M^2 - 1} \right] \cdot 3 \tag{5.8}$$

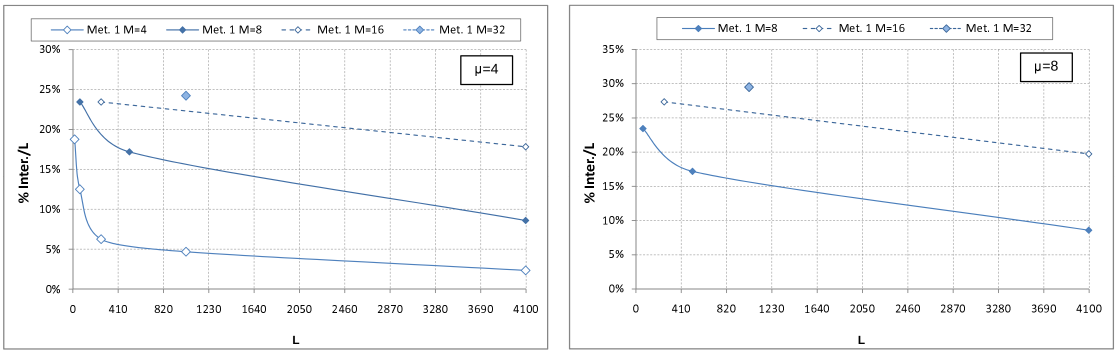
- Cuando N es impar y $N > 1$

$$W_A = W_C = \left[\frac{\frac{M}{2}(M^{(N-1)}-1)}{M^2-1}\right] \cdot 3$$
$$E_A = E_C = \left[\frac{\frac{M}{2}(M^{(N-1)}-1)}{M^2-1}\right] \cdot 3 + \frac{M^N}{2} \tag{5.9}$$

Se ha considerado que el tamaño de las IFW de una familia cualquiera de pares T-ZCZ$_1$ es el de las menores zonas de ceros laterales y centrales obtenidas de entre las M posibles SACFs y $\frac{M}{2} \cdot (M-1)$ SCCFs de la familia de códigos bajo estudio. En el caso de la función de auto-correlación, el tamaño de estas áreas W_A y E_A es el mismo para las SACFs de todos los códigos. Por contra, el tamaño de las IFWs en la suma de las correlaciones cruzadas depende de los códigos considerados. Los valores de la ecuaciones (5.8) y (5.9) corresponden al caso de evaluar los M códigos de la familia y por eso $W_A = W_C$ y $E_A = E_C$. Sin embargo, cuando se considera un menor número de usuarios simultáneos, los $\mu < M$ códigos especificados en las tablas 5.7, 5.8 y 5.9 reducen al máximo el número de interferencias y puede darse el caso $W_C > W_A$ y $E_C > E_A$.

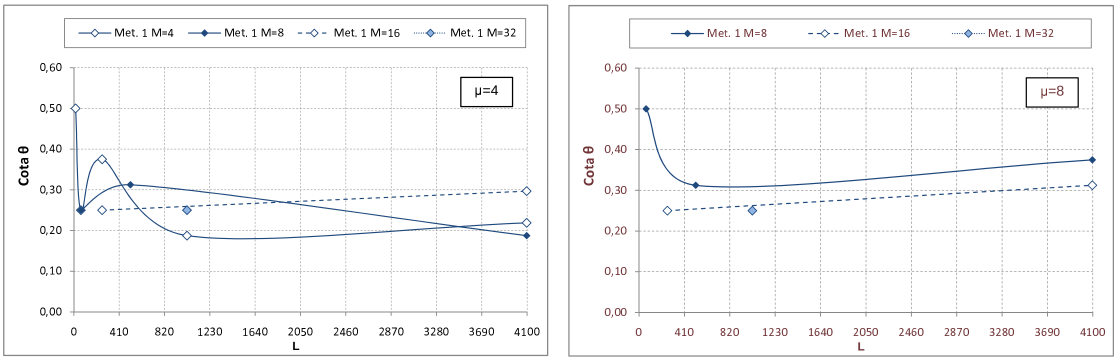
Por otro lado, revisando las interferencias que aparecen en la IW de los pares T-ZCZ$_1$, se ha observado que éstas se encuentran en las mismas posiciones para todas las SACFs de una familia cualquiera de M códigos, no ocurriendo lo mismo en el caso de la SCCF, en donde el número de interferencias depende de los códigos concretos evaluados. En cualquier caso, como ya se explicó en el capítulo 3, existe un menor riesgo de validación errónea de lóbulos laterales utilizando un código con un número elevado de interferencias de bajo valor, que con otro con pocas interferencias de alto valor; por ello en la selección de los μ códigos a emitir se prioriza reducir la cota de correlación θ en la IW y, en segundo lugar, el número de interferencias. En la figura 5.2 se muestra el porcentaje de interferencias en la longitud total de los códigos para $\mu = 4$ y $\mu = 8$ usuarios simultáneos, estando este porcentaje en todos los casos evaluados por debajo del 30 %. La figura 5.3 muestra las cotas máximas de correlación cuando se utilizan también $\mu = 4$ y $\mu = 8$ usuarios simultáneos; mientras que la figura 5.4 muestra las cotas para un rango mayor de usuarios. Puede observarse en estas figuras que la magnitud de las interferencias en la IW no excede en ningún caso la mitad del pico principal de correlación; por lo que podría distinguirse dicho pico principal de SACF aún cuando se produjese algún desajuste que implicase recibir dos códigos solapados en el tiempo con una separación mayor al tamaño de la IFW en el origen.

Si se comparan los resultados obtenidos con los de la propuesta de generación de códigos T-ZCZ$_1$ realizada por Chao Zhang et al. en [ZLH04] se tiene que, para la misma longitud de código, las interferencias están confinadas en una zona IW de menor tamaño con el método aquí propuesto. Más concretamente, el tamaño de las zonas libres de interferencias laterales E_A y E_C es mayor con este nuevo método 1 de generación que con el equivalente de Chao Zhang. Por otro lado, los valores de cota son algo menores cuando se emplean códigos T-ZCZ$_1$ con el método propuesto en este capítulo, tal y como puede observarse en la figura 5.5. Como desventaja, cabe mencionar que el rango de longitudes L disponibles en una familia de M códigos es más limitado con este método que con el de Chao Zhang.



(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.2: Porcentaje de interferencias en pares T-ZCZ$_1$ en función del número M de códigos de la familia y la longitud L de los mismos para $\mu = 4$ y $\mu = 8$ usuarios simultáneos.



(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.3: Valores de cota en la IW de pares T-ZCZ$_1$ en función del número M de códigos disponibles en la familia y de la longitud L de los mismos.

5.2.2. Influencia de la recepción parcial del código T-ZCZ$_1$ en la cota de auto-correlación

En un sistema de eco-detección, el uso de un mismo transductor como emisor y receptor implica la aparición de una *zona ciega* debido al acoplamiento de la señal emitida con la etapa de recepción. Durante la emisión el sistema de recepción está deshabilitado o saturado, por lo que un reflector que se encuentre situado a una distancia menor a la que recorre la onda durante la emisión no será detectado correctamente. En trabajos previos [HUM$^+$06, Á05] ha quedado demostrado que basta con recibir un porcentaje reducido del código emitido para obtener después de la correlación un pico principal suficientemente elevado que permita distinguirlo de los lóbulos laterales de ACF.

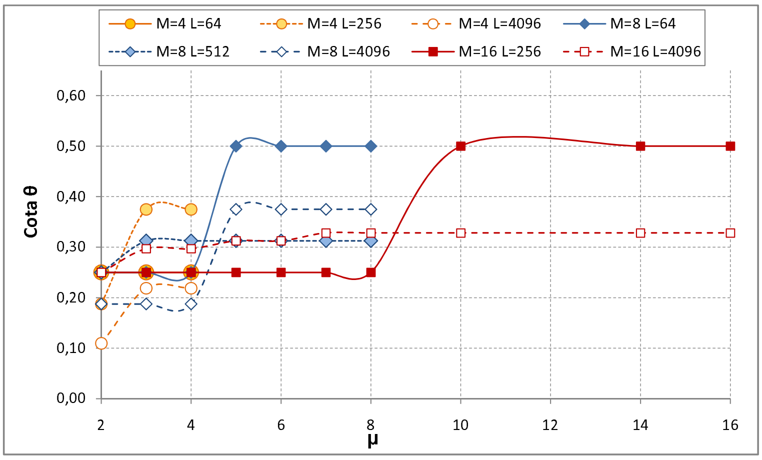
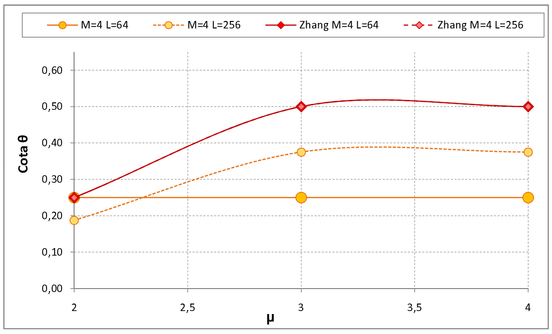
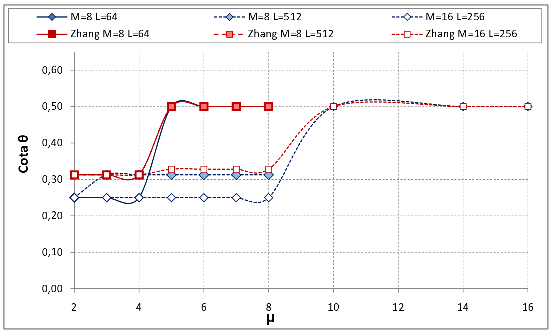


Figura 5.4: Valores de cota θ en la IW de pares T-ZCZ$_1$ en función del número M de códigos disponibles en la familia, su longitud L y el número μ de usuarios simultáneos.



(a) $M = 4$



(b) $M = 8$ y $M = 16$

Figura 5.5: Comparativa de los valores de cota θ máximos de pares T-ZCZ$_1$ obtenidos con el método 1 propuesto y el método 1 de Chao Zhang et al. [ZLH04].

En los pares T-ZCZ, la pérdida del inicio del código supone un incremento de los lóbulos laterales en la IW y la aparición de interferencias en las IFWs alrededor del origen y laterales, tal y como puede observarse en la figura 5.6 para un código T-ZCZ$_1$ de $L = 64$ bits. Obsérvese que es posible detectar los dos ecos recibidos, a pesar de que el segundo corresponde a la recepción de únicamente el 35 % del código.

La figura 5.7 muestra en línea continua la cota de auto-correlación θ_{AC} en la IW, y en línea discontinua la cota θ_{AC} en la IFW alrededor del origen (W_A) cuando se emiten $\mu = 4$ códigos simultáneos de longitud $L = 64$ y $L = 256$ bits respectivamente. En la figura 5.8 pueden observarse los resultados para una familia de $M = 8$ códigos y $\mu = 8$ usuarios simultáneos. En general, familias con un número de códigos M menor tienen valores de cota máxima más bajos. Por otro lado, se tiene que los valores de cota θ_{AC} obtenidos con estos códigos T-ZCZ$_1$ cuando se recibe sólo una parte de los mismos, no distan en exceso de los valores de cota obtenidos con otros códigos evaluados en el capítulo 3, como los Kasami.

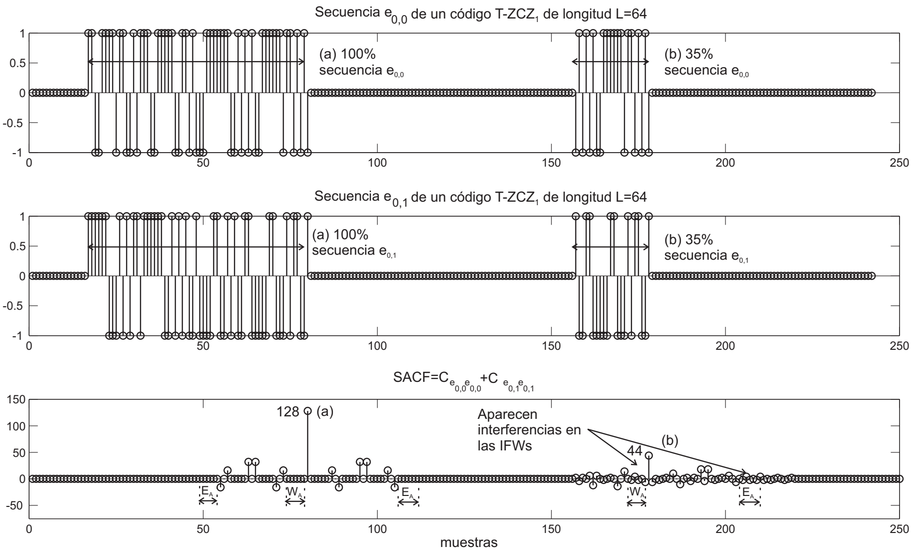
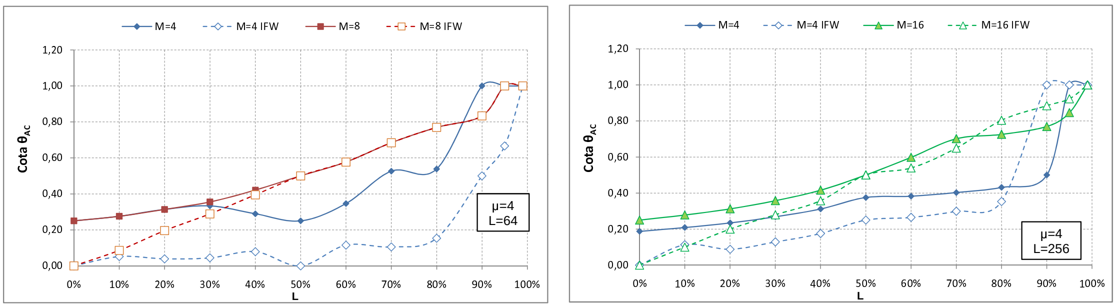


Figura 5.6: Ejemplo de SACF de un código T-ZCZ$_1$ de $L = 64$ bits cuando a) se recibe todo el código y b) se recibe únicamente el 35 %.



(a) $L = 64$ (b) $L = 256$

Figura 5.7: θ_{AC} en función del porcentaje de código perdido para $\mu = 4$ códigos T-ZCZ$_1$.

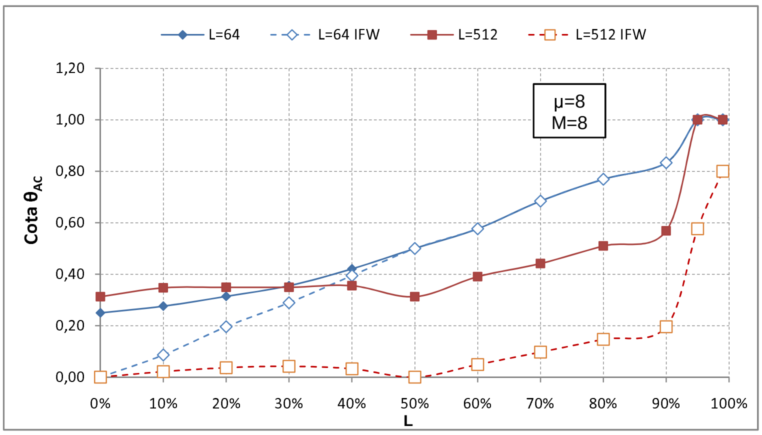


Figura 5.8: θ_{AC} en función del porcentaje de código perdido para $\mu = 8$ códigos T-ZCZ$_1$.

5.2.3. Propuesta de un generador eficiente

Puesto que los pares T-ZCZ se forman a partir de CSS es posible aprovechar los algoritmos ESSG y ESSC para reducir el número de operaciones a llevar a cabo en la generación y correlación de dichos pares T-ZCZ.

Utilizando un ESSG con una semilla cualquiera $W_N^{(m}$ y cuya entrada sea la secuencia impulsiva $\delta[\tau]$, se obtienen M salidas correspondientes a las secuencias complementarias $S_i = \{s_{i,0}, s_{i,1}, \cdots, s_{i,M-1}\}$. Estas secuencias son las que constituyen los $\frac{M}{2}$ primeros pares T-ZCZ$_1$: $e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2} = s_{i,j}$, con $0 \leq j \leq M-1$.

Los $\frac{M}{2}$ pares T-ZCZ$_1$ siguientes están formados por las secuencias de otro conjunto $S_{i'} = \{\overline{s_{i,1}}, -\overline{s_{i,0}}, \overline{s_{i,2}}, -\overline{s_{i,1}}, \cdots, \overline{s_{i,M-1}}, -\overline{s_{i,M-2}}\}$ incorrelado con el conjunto S_i. Sabiendo que $\overline{s_{i,j}}[\tau] = s_{i,j}[L_0 - 1 - \tau]$ denota colocar en orden contrario los bits de la secuencia $s_{i,j}$ de longitud L_0 y $-s_{i,j}$ negar los bits de dicha secuencia, es posible generar de modo eficiente las secuencias del conjunto S_i', y por tanto los $M/2$ pares T-ZCZ$_1$ restantes, modificando el generador ESSG. Conmutando el orden de los coeficientes $\{0, 1, \cdots, M-2, M-1\}$ que multiplican los retardos D_n de cada etapa del ESSG por los coeficientes $\{M-1, M-2, \cdots, 1, 0\}$, se obtiene un generador de secuencias reflejadas $\overline{s_{i,j}}$, al que se ha denominado M-$\overline{\text{ESSG}}$ y que requiere los mismos recursos y número de operaciones que el M-ESSG original. Negando las salidas pares del generador M-$\overline{\text{ESSG}}$ se obtiene un generador eficiente para el conjunto S_i', o lo que es lo mismo, se obtienen las secuencias $e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2} = (-1)^{j+1} \cdot \overline{s_{i,j}}$.

En la figura 5.9 se muestra la arquitectura, basada en el ESSG y $\overline{\text{ESSG}}$, del generador eficiente de códigos T-ZCZ$_1$ propuesto (ETZG$_1$, *Efficient Three Zero correlation zones codes Generator*; el sub-índice 1 denota que los pares T-ZCZ se generaron empleando el método 1 y la ausencia de tilde que se usó un único CSS).

5.2.4. Propuesta de un correlador eficiente

Teniendo en cuenta que, tal y como se ha visto en el capítulo anterior, la respuesta impulsiva de un filtro acoplado a una secuencia determinada es la versión reflejada en el tiempo de esa secuencia, puede considerarse que el ETZG$_1$ es un filtro acoplado a las versiones reflejadas en el tiempo $\{(\overline{e_{k,0}}, \overline{e_{k,1}}); 0 \leq k \leq M-1\}$. Por tanto, modificando el orden de los retardos en el ETZG$_1$ para que en vez de generar las versiones directas genere las reflejadas, se obtendrá un correlador acoplado a estas secuencias directas. Este nuevo filtro, mostrado en la figura 5.10, puede considerarse como un correlador eficiente de códigos T-ZCZ$_1$ (ETZC$_1$, *Efficient Three Zero correlation zones codes Correlator*) y realiza la correlación simultánea de una señal de entrada $r[\tau]$ con las secuencias de los M pares de una familia T-ZCZ$_1$.

La correlación de la señal de entrada $r[\tau]$ con las M secuencias $\{s_{i,j}; 0 \leq j \leq M-1\}$ de los primeros $M/2$ pares de la familia T-ZCZ$_1$ puede realizarse mediante un ESSC con valores

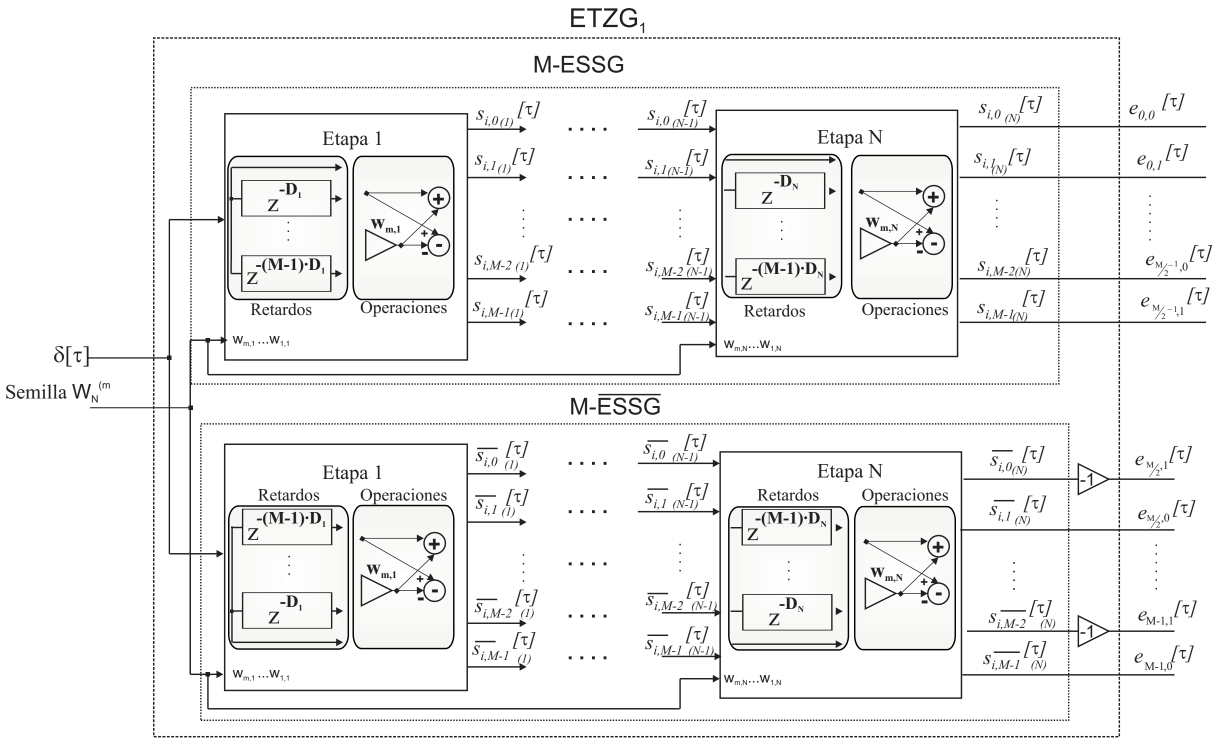


Figura 5.9: Diagrama de bloques del generador eficiente propuesto para códigos T-ZCZ$_1$ (ETZG$_1$).

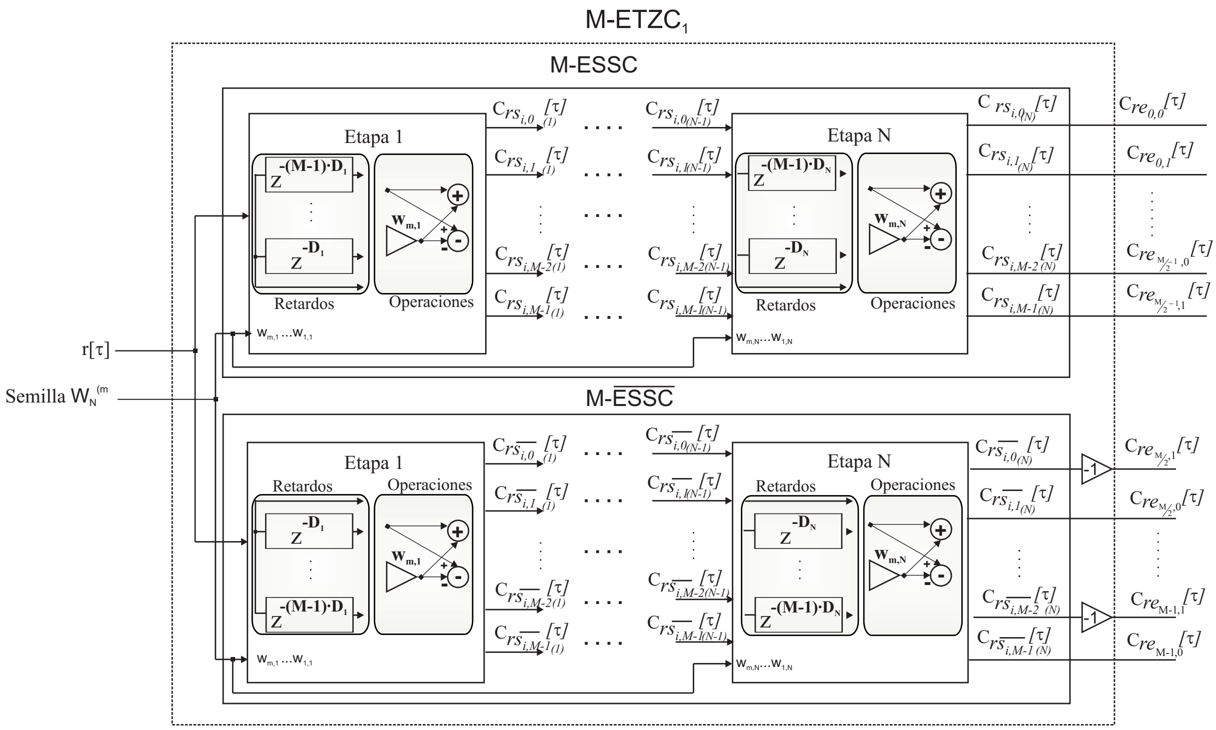


Figura 5.10: Diagrama de bloques del correlador eficiente propuesto para códigos T-ZCZ$_1$ (ETZC$_1$).

de semilla $W_N^{(m}$ correspondientes al conjunto S_i. Las correlaciones $C_{r,s_{i,j}}[\tau]$ obtenidas son, en definitiva, $C_{r,s_{i,j}} = C_{r,e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2}}$, con $0 \leq j \leq M-1$. Por otro lado, cambiando el orden de los coeficientes $\{M-1, M-2, \cdots, 1, 0\}$ que multiplican los retardos D_n de cada etapa del ESSC por los coeficientes $\{0, 1, \cdots, M-2, M-1\}$, se obtiene un correlador M-$\overline{\text{ESSC}}$ acoplado a las secuencias reflejadas $\overline{s_{i,j}}$, y que tiene las mismas necesidades computacionales que el M-ESSC original. Las salidas pares del correlador M-$\overline{\text{ESSC}}$ se multiplican por -1 para obtener $C_{r,\overline{s_{i,j}}} = (-1)^{j+1} \cdot C_{r,e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2}}$.

Resumiendo, la señal de entrada $r[\tau]$ es procesada en los correladores M-ESSC y M-$\overline{\text{ESSC}}$, negando las salidas pares de este último, para conseguir las correlaciones $\{(C_{r,e_{k,0}}, C_{r,e_{k,1}}); 0 \leq k \leq M-1\}$ con los M pares de una familia cualquiera de códigos T-ZCZ$_1$. Un ejemplo de un ETCZ$_1$ asociado a una familia de $M = 4$ códigos T-ZCZ$_1$ puede observarse en la figura 5.11. Obsérvese que la primera etapa de retardos puede ser compartida entre los correladores ESSC y $\overline{\text{ESSC}}$ para reducir las necesidades de memoria.

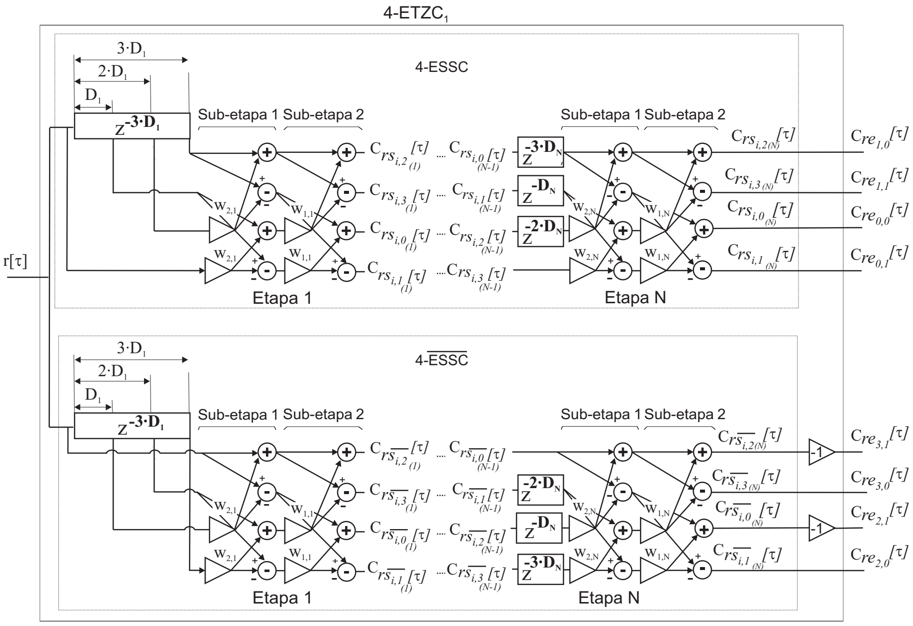


Figura 5.11: Ejemplo de un 4-ETZC$_1$ para códigos T-ZCZ$_1$.

La razón de haber añadido el calificativo *eficiente* al correlador ETZC$_1$ propuesto se debe a que realiza menos operaciones que uno directo convencional. Con objeto de obtener en el modelo directo la correlación de la señal de entrada con los M pares de una familia de códigos T-ZCZ$_1$ de forma simultánea, el bloque encargado de realizar las sumas y restas de la correlación tiene que ser replicado $2 \cdot M$ veces, para detectar las secuencias de los M pares individualmente; mientras que el buffer que almacena los datos de entrada puede ser compartido ahorrando de este modo posiciones de memoria (recuérdese la figura 4.1). La tabla 5.10 muestra una comparativa de las necesidades computacionales de ambas opciones, el correlador directo tradicional y el ETZC$_1$, para la correlación de la señal de entrada con las secuencias de una familia T-ZCZ$_1$. Las operaciones de multiplicación que aparecen en dicha tabla pueden implementarse en la práctica como sumas y restas.

Para una comparación más inmediata, en la figura 5.12 se representa el número de operaciones y bits consumidos por el correlador directo y el ETZC$_1$, cuando se trata de detectar en la señal recibida la presencia de alguno de los $M = 8$ códigos de longitud L de una familia T-ZCZ$_1$, con un ancho de bus de datos de entrada de $DW = 8$ bits, y siendo $N_{SM} = O_f = 1$. Obsérvese la importante reducción en el número de operaciones a llevar a cabo en la correlación cuando se usa el ETZC$_1$.

Implementación	Productos	Sumas	Bits de Memoria
Correlador directo serie	$2M^{N+1}$	$2M \cdot (M^N - 1)$	$N_{SM} \cdot O_f \cdot DW \cdot M^N + 2M^{N+1} + 2M \cdot (DW + mN)$
M-ETZC$_1$	$M \cdot m \cdot N + \frac{M}{2}$	$2M \cdot m \cdot N$	$2N_{SM} \cdot O_f \cdot DW \cdot (M-1)M^{N-1} + N_{SM} \cdot O_f \cdot (M^2 - M)\left[DW\frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right]$

Tabla 5.10: Comparativa de las necesidades computacionales de un correlador directo serie y el ETZC$_1$ propuesto, para la correlación de la señal de entrada con las secuencias de los M pares de una familia T-ZCZ$_1$ de longitud $L = M^N$.

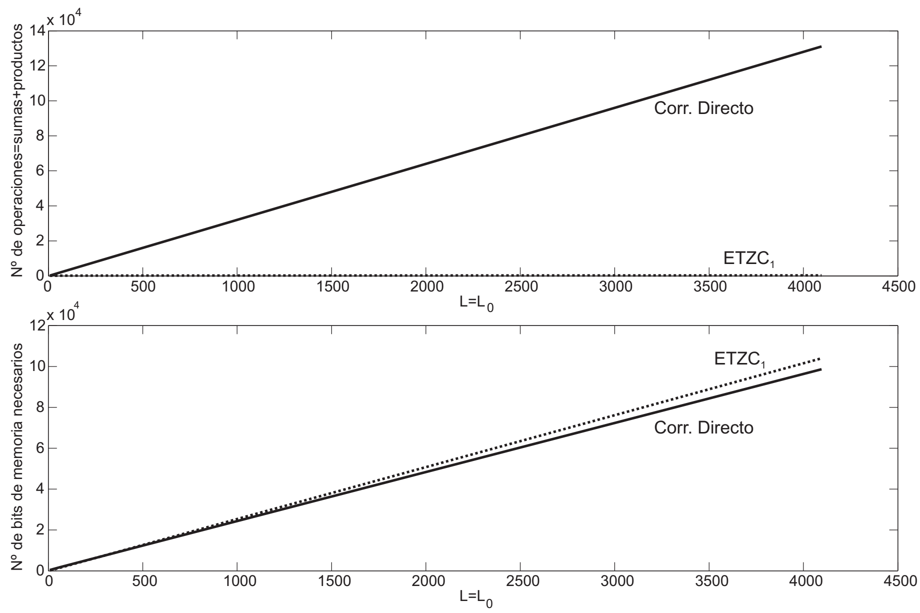


Figura 5.12: Comparativa del número de operaciones y recursos de memoria requeridos por un correlador directo serie y el ETZC$_1$ propuesto, para la detección de $M = 8$ usuarios simultáneos con $DW = 8$.

5.2.5. Propiedades de correlación y propuesta de algoritmos eficientes para códigos T-ZCZ$_{1'}$ obtenidos a partir de $\Delta^{(M,M\cdot L_0}$

Los códigos T-ZCZ$_{1'}$ se obtienen mediante el método 1 usando M UCSS. Se comenta aquí brevemente esta opción sin entrar en demasiado detalle puesto que, aunque es posible obtener un correlador ETZC$_{1'}$ asociado a estos códigos que reduzca el número de operaciones en comparación con uno directo, es menos eficiente en cuanto a operaciones a realizar y consumo de memoria que el ETZC$_1$ propuesto para códigos T-ZCZ$_1$ obtenidos a partir de un único CSS, según puede observarse en la figura 5.13 para una familia con $M = 4$ códigos y un ancho de bus de datos $DW = 8$ bits.

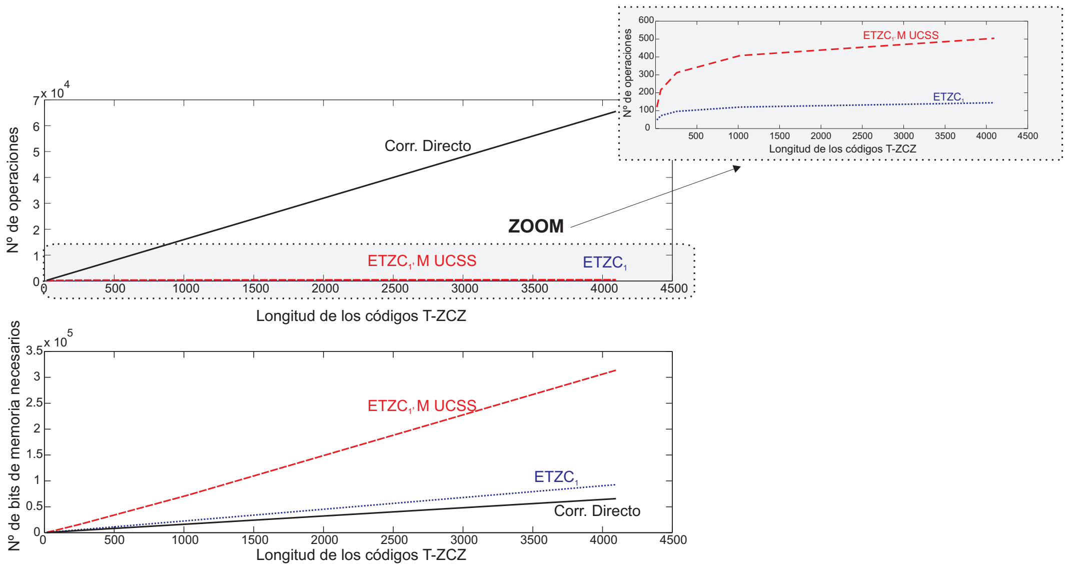


Figura 5.13: Comparativa del número de operaciones y recursos de memoria requeridos para la correlación eficiente de pares T-ZCZ generados con el método 1 a partir de un único CSS, o a partir de M UCSS.

Teniendo en cuenta que los pares T-ZCZ$_{1'}$ están formados por las secuencias $E_0 = (e_{0,0}, e_{0,1}) = ([s_{0,0}|s_{1,0}|\cdots|s_{M-1,0}], [s_{0,1}|s_{1,1}|\cdots|s_{M-1,1}])$, $\cdots$, $E_{M/2-1} = (e_{M/2-1,0}, e_{M/2-1,1}) = ([s_{0,M-2}|s_{1,M-2}|\cdots|s_{M-1,M-2}], [s_{0,M-1}|s_{1,1M-1}|\cdots|s_{M-1,M-1}])$, $\cdots$, $E_{M/2} = (e_{M/2,0}, e_{M/2,1}) = ([\overline{s_{M-1,1}}|\overline{s_{M-2,1}}|\cdots|\overline{s_{0,1}}], [-\overline{s_{M-1,0}}| - \overline{s_{M-2,0}}|\cdots| - \overline{s_{0,0}}])$, $\cdots$, $E_{M-1} = (e_{M-1,0}, e_{M-1,1}) = ([\overline{s_{M-1,M-1}}|\overline{s_{M-2,M-1}}|\cdots|\overline{s_{0,M-1}}], [-\overline{s_{M-1,M-2}}| - \overline{s_{M-2,M-2}}|\cdots| - \overline{s_{0,M-2}}])$; se necesitan M ESSCs para obtener la correlación de la señal de entrada $r[\tau]$ con cada una de las M^2 secuencias $\{s_{i,j}; 0 \leq i,j \leq M-1\}$ de la matriz $\Delta^{(M,M\cdot L_0}$; y M $\overline{\text{ESSCs}}$ para tener la correlación con las M^2 secuencias reflejadas $\{\overline{s_{i,j}}; 0 \leq i,j \leq M-1\}$. Los resultados de correlación con los M pares T-ZCZ$_{1'}$ se obtienen sumando las salidas j-ésimas $C_{r,s_{i,j}}$ y $C_{r,\overline{s_{i,j}}}$ de los correladores ESCCs y $\overline{\text{ESSCs}}$ con diferentes desfases, que dependen del orden en el que las secuencias j-ésimas $s_{i,j}$ y $\overline{s_{i,j}}$ fueron concatenadas para formar dichos pares T-ZCZ$_{1'}$. Además, es necesario negar el resultado de los sumatorios de las correlaciones $C_{r,\overline{s'_{i,j}}}$ cuando j es par, para obtener las correlaciones $\{C_{r,e_{k,1}}; M/2 \leq k \leq M-1\}$. El diagrama de bloques de la figura 5.14 muestra de forma esquematizada y genérica el correlador ETZC$_{1'}$ descrito. El número de operaciones y recursos de memoria necesarios para la implementación del correlador, en comparación con uno directo, aparecen en la tabla 5.11. Obsérvese que, en el ETZC$_{1'}$, las necesidades computacionales corresponden a las de los M ESSCs del paso 1, más las asociadas a los bloques de retardo del paso 2 y a los sumadores y multiplicadores del paso 3 (estos multiplicadores conllevan únicamente un cambio de signo y pueden implementarse como restas).

Implementación	Productos	Sumas	Bits de Memoria
Correlador directo serie	$2M^{N+2}$	$2M(M^{N+1}-1)$	$N_{SM} \cdot O_f \cdot DW \cdot M^{N+1} + 2M^{N+2} + 2M \cdot (DW + mN + m)$
M-ETZC$_{1'}$	$M^2 mN + \frac{M}{2}$	$2M^2 mN + 2(M^2 - M)$	$2N_{SM} \cdot O_f \cdot DW \cdot (M-1)M^N + N_{SM} \cdot O_f \cdot (M^3 - M^2)\left[DW \frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right] + N_{SM} \cdot O_f \cdot (DW + mN) \cdot (M^2 - M) \cdot M^N$

Tabla 5.11: Comparativa de las necesidades computacionales de un correlador directo serie y el ETZC$_{1'}$ propuesto, para la correlación de la señal de entrada con las secuencias de los M pares de una familia T-ZCZ$_{1'}$ de longitud $L = M^{N+1}$.

Las propiedades de correlación de los códigos T-ZCZ$_{1'}$ son parecidas a las propiedades de los códigos T-ZCZ$_1$. El porcentaje de interferencias en la IW es similar, como puede observarse en la figura 5.15, como también los valores de cota obtenidos en la IW (véase la figura 5.16).

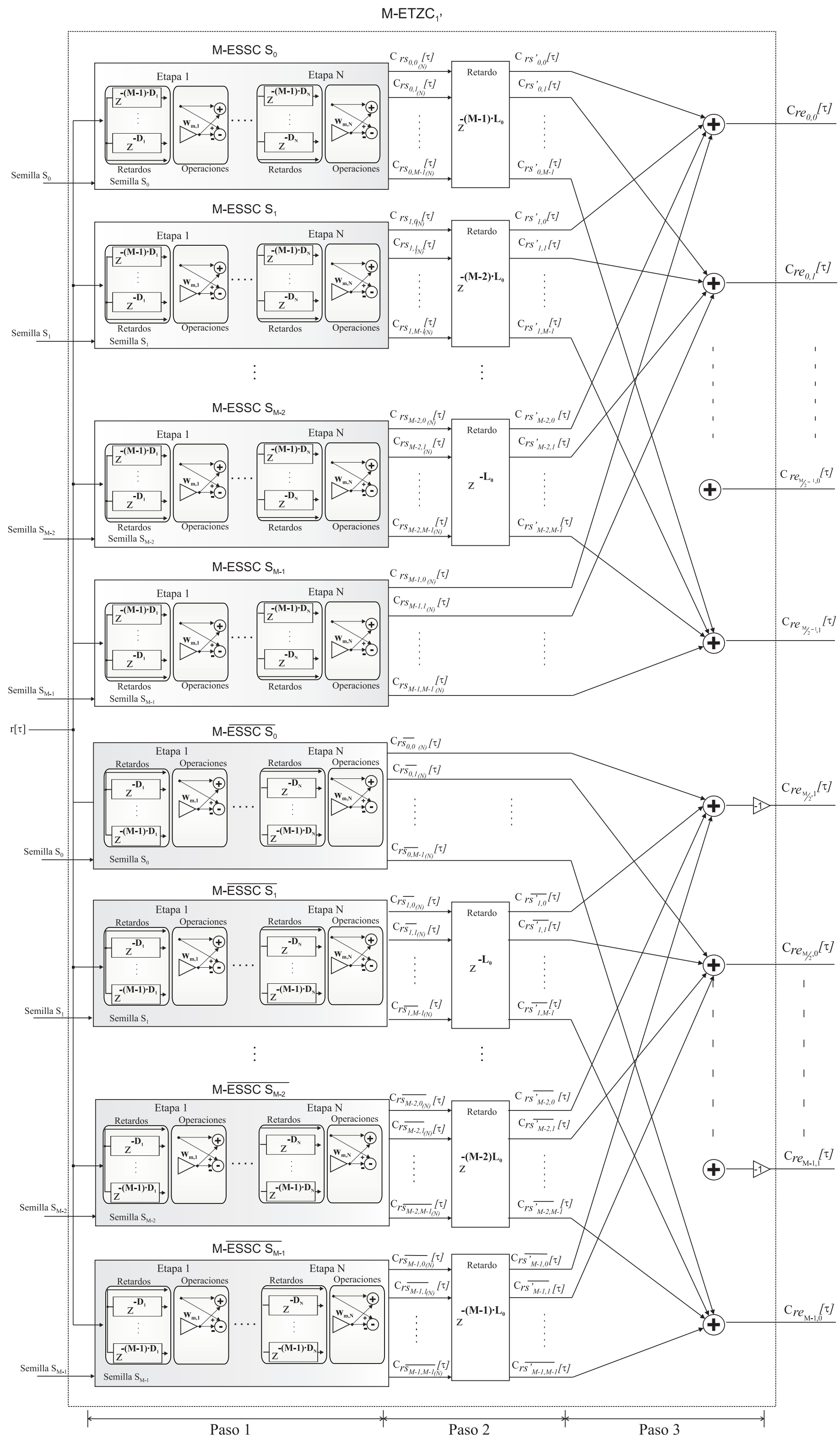
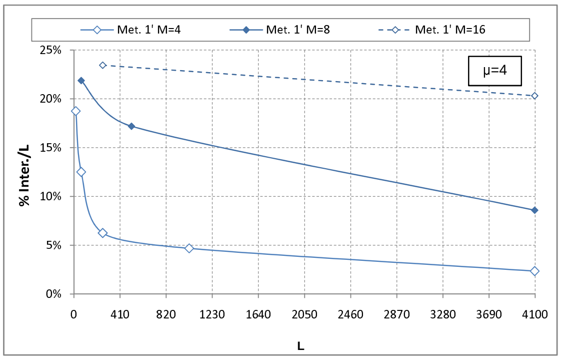
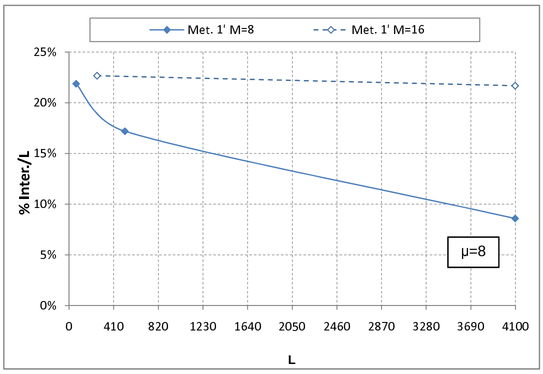


Figura 5.14: Diagrama de bloques del correlador eficiente propuesto para códigos T-ZCZ$_{1'}$ generados con el método 1 a partir de M UCSS (ETZC$_{1'}$).



(a) $\mu = 4$



(b) $\mu = 8$

Figura 5.15: Porcentaje de interferencias en pares T-ZCZ$_{1'}$ en función del número M de códigos de la familia y la longitud L de los mismos para $\mu = 4$ y $\mu = 8$ usuarios simultáneos.

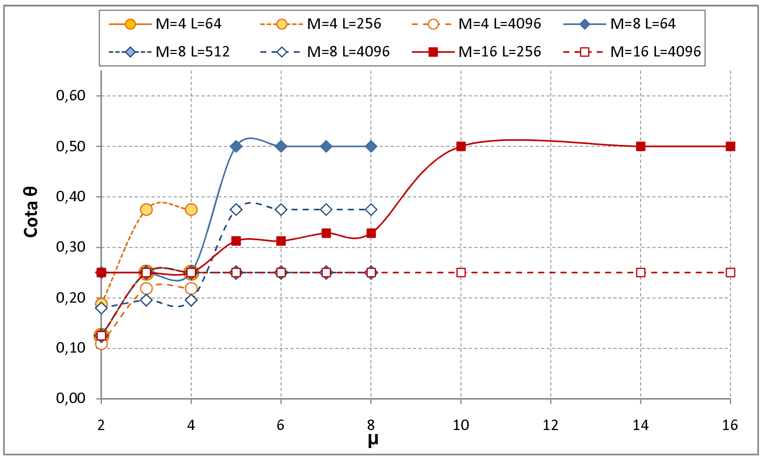


Figura 5.16: Valores de cota θ en la IW de pares T-ZCZ$_{1'}$ en función del número M de códigos disponibles en la familia, su longitud L y el número μ de usuarios simultáneos.

5.3. Caracterización del método 2 usando $\Delta^{(M,M\cdot L_0}$

5.3.1. Estudio de las propiedades de correlación

Cuando se elige el método 2 para la generación de códigos T-ZCZ, son los T-ZCZ$_{2'}$ obtenidos a partir de $\Delta^{(M,M\cdot L_0}$ los que derivan en una implementación más eficiente del correlador utilizado en su detección. Por ello, en esta sección se dedica un mayor espacio al estudio de los pares T-ZCZ$_{2'}$, realizando un breve apunte comparativo con los pares T-ZCZ$_2$ al final de la misma.

Las tablas 5.12 a 5.14 resumen las propiedades de correlación de dichos códigos T-ZCZ$_{2'}$. A grandes rasgos, puede concluirse que los valores de cota máximos en la IW no superan la mitad del valor del pico principal de SACF, y que en la IW aparecen también cadenas de ceros. Resulta posible, por tanto, detectar correctamente el código emitido incluso cuando ecos de un mismo código se reciben con una separación superior a la IFW.

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
8	2	0.25	0.25	0	0, 1	2	2	66.67 %	0 %
	3	0.5	0.25	0.5	0, 1, 2				100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
32	2	0.375	0.375	0	0, 2	9	9	30.77 %	0 %
	3	0.375	0.375	0.375	0, 1, 2				46.15 %
	4	0.375	0.375	0.375	0, 1, 2, 3				46.15 %
128	2	0.1875	0.1875	0	0, 1	38	38	23.53 %	0 %
	3	0.375	0.1875	0.375	0, 1, 2				31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %
512	2	0.2188	0.2188	0	0, 2	153	153	11.71 %	0 %
	3	0.2188	0.2188	0.2188	0, 1, 2				17.56 %
	4	0.2188	0.2188	0.2188	0, 1, 2, 3				17.56 %
2048	2	0.1094	0.1094	0	0, 3	614	614	8.79 %	0 %
	3	0.2188	0.1094	0.2188	0, 1, 2				11.72 %
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %
8192	2	0.1484	0.1484	0	0, 3	2457	2457	4.39 %	0 %
	3	0.1484	0.1484	0.1484	0, 1, 2				6.59 %
	4	0.1484	0.1484	0.1484	0, 1, 2, 3				6.59 %

Tabla 5.12: Estudio de la zona con interferencias de familias T-ZCZ$_{2'}$ con $M = 4$ códigos.

M = 8

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
32	2	0.3125	0.3125	0	0, 1	4	4	43.38 %	0 %
	3	0.3125	0.3125	0.25	0, 1, 6				60.87 %
	4	0.3125	0.3125	0.25	0, 1, 6, 7				60.87 %
	5	0.5	0.3125	0.5	0, 1, 2, 3, 4				60.87 %
	6	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5				60.87 %
	7	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6				60.87 %
	8	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7				60.87 %
256	2	0.125	0.125	0	0, 4	33	33	22.22 %	0 %
	3	0.25	0.25	0.25	0, 3, 4				29.63 %
	4	0.25	0.25	0.25	0, 3, 4, 7				29.63 %
	5	0.375	0.375	0.375	0, 1, 2, 3, 4				29.63 %
	6	0.375	0.375	0.375	0, 1, 2, 3, 4, 5				29.63 %
	7	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6				29.63 %

Continuación de la Tabla 5.13

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
	8	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7				29.63 %
2048	2	0.1797	0.1797	0	0, 5	268	268	15.88 %	0 %
	3	0.1953	0.1953	0.1523	0, 1, 7				23.30 %
	4	0.1953	0.1953	0.1523	0, 1, 6, 7				23.30 %
	5	0.375	0.1953	0.375	0, 1, 2, 3, 4				23.30 %
	6	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5				23.30 %
	7	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5, 6				23.30 %
	8	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5, 6, 7				23.30 %
16384	2	0.2188	0.2188	0	0, 5	2145	2145	7.97 %	0 %
	3	0.2188	0.2188	0.1162	0, 3, 4				11.61 %
	4	0.2188	0.2188	0.1162	0, 3, 4, 7				11.61 %
	5	0.2188	0.2188	0.2188	0, 1, 2, 3, 4				12.97 %
	6	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5				12.97 %
	7	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5, 6				12.97 %
	8	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5, 6, 7				12.97 %

Tabla 5.13: Estudio de la zona con interferencias de familias T-ZCZ$_{2'}$ con $M = 8$ códigos.

M = 16

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
128	2	0.3125	0.3125	0	2, 3	8	8	37.84 %	0 %
	3	0.3125	0.3125	0.125	2, 3, 12				54.05 %
	4	0.3125	0.3125	0.125	2, 3, 12, 13				54.05 %
	5	0.3281	0.3281	0.25	0, 1, 6, 7, 10				52.25 %
	6	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11				52.25 %
	7	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11, 12				52.25 %
	8	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11, 12, 13				52.25 %
	10	0.5	0.3281	0.5	2, 3, 4, 5, 6, 7, 8, 9, 10, 11				54.05 %
	14	0.5	0.3281	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				54.05 %
	16	0.5	0.3281	0.5	0, 1, 2, $\cdots$, 15				54.05 %
2048	2	0.375	0.125	0	0, 8	129	129	20.12 %	0 %
	3	0.375	0.375	0.0762	0, 7, 8				29.96 %
	4	0.375	0.375	0.0762	0, 7, 8, 15				29.96 %
	5	0.375	0.375	0.2031	0, 3, 5, 6, 8				29.96 %
	6	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11				29.96 %
	7	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11, 13				29.96 %
	8	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11, 13, 14				29.96 %

Continuación de la Tabla 5.14

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
	10	0.375	0.25	0.25	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				29.96 %
	14	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				29.96 %
	16	0.375	0.375	0.375	0, 1, 2, $\cdots$, 15				29.96 %

Tabla 5.14: Estudio de la zona con interferencias de familias T-ZCZ$_{2'}$ con $M = 16$ códigos.

Del análisis de las tablas anteriores puede deducirse la expresión (5.10), que permite calcular el tamaño de las zonas libres de interferencias en la SACF y SCCF para un número M de pares T-ZCZ$_{2'}$ concreto y en función del número N de etapas o iteraciones del algoritmo de generación de los CSS (recuérdese que la longitud de dichos CSS es $L_0 = M^N$, mientras que la de los pares T-ZCZ$_{2'}$ es $L = \frac{M^{N+1}}{2}$).

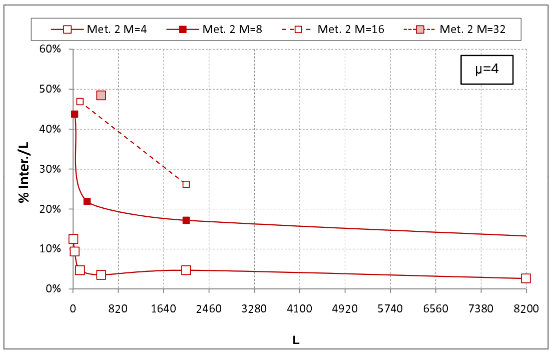
$$\begin{array}{lll} n = 1 & \rightarrow & X_{(1)} = \frac{M}{2} \\ n = 2 & \rightarrow & X_{(2)} = \frac{M^2}{2} + 1 \\ n \geq 3 & \rightarrow & X_{(n)} = \frac{M^n}{2} + M^{n-2} + X_{(n-2)} \end{array} \tag{5.10}$$

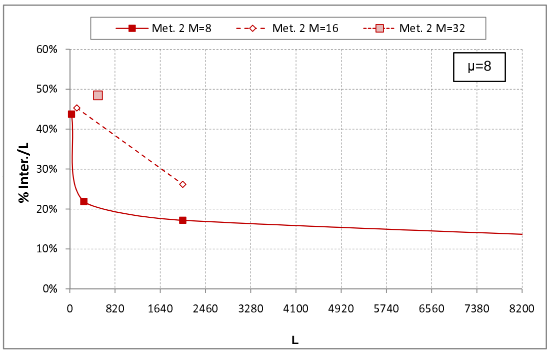
En la iteración $n = N$ se obtiene en $X_{(N)}$ el tamaño de las zonas libres de interferencias $W_A = W_C = E_A = E_C = X_{(N)}$. Obsérvese que con este método el tamaño de las IFWs centrales y laterales coincide para todas las longitudes L de código.

La figura 5.17 muestra el porcentaje de valores no nulos respecto a la longitud total del código. Puede observarse que el porcentaje de interferencias correspondiente a códigos T-ZCZ$_{2'}$ de longitud $L = \frac{M^{N+1}}{2}$ es similar al porcentaje obtenido con códigos T-ZCZ$_1$ con longitud $L = M^N$. En definitiva se necesitan códigos T-ZCZ$_{2'}$ $\frac{M}{2}$ veces más largos que los T-ZCZ$_1$ para tener el mismo porcentaje de ceros.

Del mismo modo, las figuras 5.18 y 5.19 representan los valores de cota θ máximos en la IW en función de la longitud L y número μ de usuarios simultáneos. La comparación de estas cotas con las obtenidas con códigos T-ZCZ$_1$ no es directa, puesto que los códigos tienen distintas longitudes (para un código T-ZCZ$_{2'}$ cualquiera, el código T-ZCZ$_1$ de longitud más próxima tiene el doble de bits que el anterior). En cualquier caso, los valores de cota no distan demasiado y en ambos métodos son $\theta \leq 0.5$.

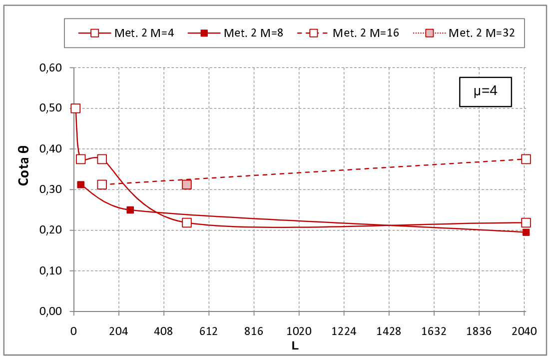
Por último, contrastando los resultados de correlación conseguidos con los códigos T-ZCZ$_{2'}$ aquí propuestos con los T-ZCZ$_2$ de Chao Zhang et al. [ZLH04], se tiene que las áreas libres de interferencias de los T-ZCZ$_{2'}$ descritos en este capítulo son algo mayores, aunque esta diferencia va decayendo conforme aumenta el número M de códigos de la familia. También los valores de cota θ son algo menores con el método 2 a partir $\Delta^{(M,M\cdot L_0}$ propuesto en esta tesis, según puede observarse en la figura 5.20.

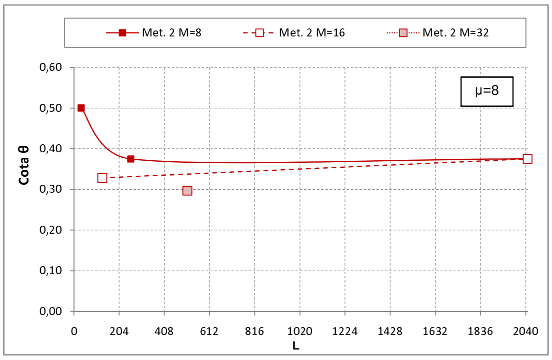




(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.17: Porcentaje de interferencias en pares T-ZCZ$_{2'}$ en función del número M de códigos de la familia y la longitud L de los mismos para $\mu = 4$ y $\mu = 8$ usuarios simultáneos.





(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.18: Valores de cota en la IW de pares T-ZCZ$_{2'}$ en función del número M de códigos disponibles en la familia y de la longitud L de los mismos.

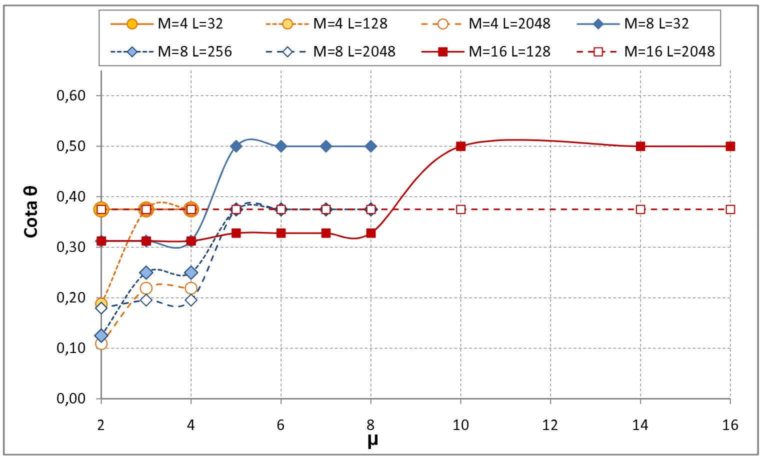


Figura 5.19: Valores de cota θ en la IW de pares T-ZCZ$_{2'}$ en función del número M de códigos disponibles en la familia, su longitud L y el número μ de usuarios simultáneos.

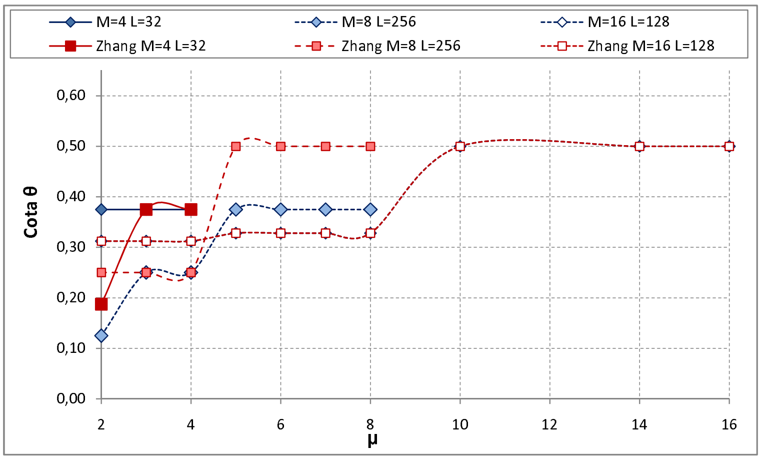
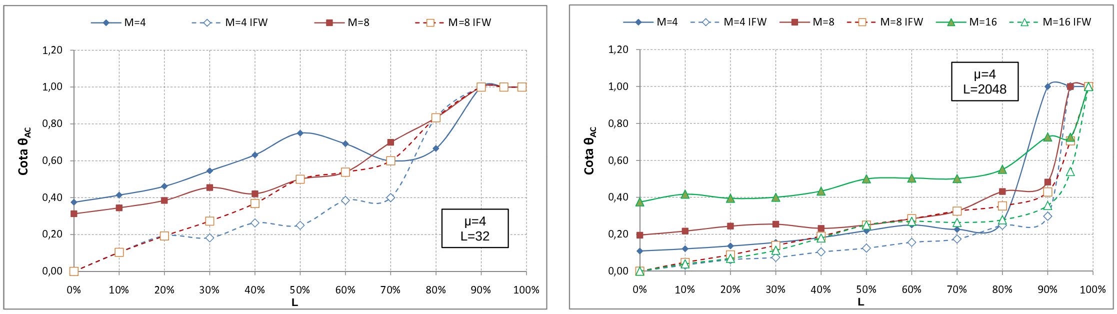


Figura 5.20: Comparativa de los valores de cota θ máximos de pares T-ZCZ$_{2'}$ obtenidos con el método 2 propuesto y el método 2 de Chao Zhang [ZLH04].

5.3.2. Influencia de la recepción parcial del código T-ZCZ$_{2'}$ en la cota de auto-correlación

Al igual que ocurre con los códigos T-ZCZ$_1$, cuando se pierde el inicio de las secuencias de un par T-ZCZ$_{2'}$ aparecen interferencias en la IFW y se incrementan las ya existentes en la IW, siendo más acusado este hecho para familias con un número M elevado de códigos. Estos resultados están en sintonía con los obtenidos para macro-secuencias de CSS en la sección 3.3.2 del capítulo 3. Se comentaba ahí que los CSS concatenados o entrelazados eran más robustos ante la pérdida de bits cuando el número M de secuencias del conjunto es reducido. Aunque la agrupación de bits de las macro-secuencias es distinta a la de las parejas T-ZCZ aquí propuestas, ambas derivan de CSS y comparten ciertas propiedades propias de estos códigos. Un ejemplo es el comportamiento parecido que presenta la función de auto-correlación de ambas familias cuando se recibe solamente una parte del código.

La figura 5.21 representa los valores de cota de SACF θ_{AC} en función del porcentaje de código pérdido para parejas de longitud $L = 32$ y $L = 2048$ bits. Obsérvese que, en sistemas de eco-detección, podrían utilizarse códigos largos sin que ello supusiese un incremento excesivo del tamaño de la zona ciega. Por ejemplo, para un par de $L = 2048$ bits es posible discriminar el pico principal de los laterales aun cuando sólo se recibe un 10 % del mismo.



(a) $L = 32$ (b) $L = 2048$

Figura 5.21: θ_{AC} en función del porcentaje de código perdido para $\mu = 4$ códigos T-ZCZ$_{2'}$.

5.3.3. Propuesta de un correlador eficiente

Para mayor brevedad, se muestra sólo la arquitectura propuesta para el correlador ETZC$_{2'}$. En aquellas aplicaciones donde resulte necesario generar los códigos T-ZCZ$_{2'}$ dinámicamente, puede obtenerse un generador eficiente ETZG$_{2'}$ invirtiendo el orden de los retardos del ETZC$_{2'}$ según se indica en [Pop99].

La secuencia $e_{j,0}$ de un par T-ZCZ$_{2'}$ cualquiera está formada por la concatenación de las j-ésimas secuencias de los CSS $\{S_i; 0 \leq i \leq \frac{M}{2} - 1\}$, mientras que en la secuencia $e_{j,1}$ se concatenan las j-ésimas secuencias de los CSS $\{S_i; \frac{M}{2} \leq i \leq M-1\}$; todos estos CSS están incorrelados y han sido obtenidos con semillas $(W d_N^{(m} + \frac{L_0}{M} \cdot i) \bmod L_0$. Sabiendo esto, las ACFs de las secuencias $e_{j,0}$ y $e_{j,1}$ pueden expresarse en función de las correlaciones con las secuencias $s_{i,j}$ que las componen según:

$$
\begin{aligned}
&C_{e_{j,0},e_{j,0}}[\tau] = \sum_{l=0}^{L-1-\tau} e_{j,0}[l]e_{j,0}[l+\tau] = \\
&= \sum_{l=0}^{L_0-1-\tau} (s_{0,j}[l]s_{0,j}[l+\tau] + s_{0,j}[l]s_{1,j}[l-L_0+\tau] + \cdots + s_{0,j}[l]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ s_{1,j}[l-L_0]s_{0,j}[l+\tau] + s_{1,j}[l-L_0]s_{1,j}[l-L_0+\tau] + \cdots + s_{1,j}[l-L_0]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ \cdots + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{0,j}[l+\tau] + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{1,j}[l-L_0+\tau] + \\
&+ \cdots + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau]) = \\
&= \sum_{i=0}^{\frac{M}{2}-1} C_{s_{i,j},s_{i,j}}[\tau] + \sum_{i=0}^{\frac{M}{2}-1} \sum_{\substack{h=0 \\ h\neq i}}^{\frac{M}{2}-1} C_{s_{i,j},s_{i,h}}[\tau-(h-i)L_0]
\end{aligned}
\tag{5.11}
$$

$$\begin{aligned}
&C_{e_{j,1},e_{j,1}}[\tau] = \sum_{l=0}^{L-1-\tau} e_{j,1}[l]e_{j,1}[l+\tau] = \\
&= \sum_{l=0}^{L_0-1-\tau} (s_{\frac{M}{2},j}[l]s_{\frac{M}{2},j}[l+\tau] + s_{\frac{M}{2},j}[l]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + s_{\frac{M}{2},j}[l]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ s_{\frac{M}{2}+1,j}[l-L_0]s_{\frac{M}{2},j}[l+\tau] + s_{\frac{M}{2}+1,j}[l-L_0]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + \\
&+ s_{\frac{M}{2}+1,j}[l-L_0]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \cdots + s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2},j}[l+\tau] + \\
&+ s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau]) = \\
&= \sum_{i=\frac{M}{2}}^{M-1} C_{s_{i,j},s_{i,j}}[\tau] + \sum_{i=\frac{M}{2}}^{M-1} \sum_{\substack{h=\frac{M}{2} \\ h\neq i}}^{M-1} C_{s_{i,j},s_{i,h}}[\tau-(h-i)L_0]
\end{aligned} \tag{5.12}$$

Las CCFs de las secuencias $(e_{j,0}, e_{j,1})$ de un código E_j pueden expresarse en función de las correlaciones con las secuencias $s_{i,j}$ de modo similar a como se ha hecho en (5.11) y (5.12). En general, la correlación de una señal cualquiera $r[\tau]$ con las secuencias de un par E_j viene dada por (5.13), en donde las correlaciones $C_{r,s_{i,j}}$ pueden realizarse de modo eficiente mediante ESSCs, reduciendo de este modo el número total de operaciones a llevar a cabo[2]. El correlador obtenido se ha denotado como ETZC$_{2'}$, y permite la correlación simultánea de la señal de entrada con las secuencias de los M pares de una familia T-ZCZ$_{2'}$.

$$\begin{aligned}
C_{r,e_{j,0}} &= C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau-L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau-(\tfrac{M}{2}-1)L_0] \\
C_{r,e_{j,1}} &= C_{r,s_{\frac{M}{2},j}}[\tau] + C_{r,s_{\frac{M}{2}+1,j}}[\tau-L_0] + \cdots + C_{r,s_{M-1,j}}[\tau-(\tfrac{M}{2}-1)L_0]
\end{aligned} \tag{5.13}$$

En la figura 5.22 se representa la arquitectura, dividida en pasos, del ETZC$_{2'}$ propuesto. En el paso 1 los $\frac{M}{2}$ ESSCs situados en la zona superior realizan la correlación eficiente de la señal de entrada $r[\tau]$ con las secuencias $\{s_{i,j}; 0 \le i \le \frac{M}{2}; 0 \le j \le M-1\}$, que son las que forman las secuencias $e_{j,0}$. Los $\frac{M}{2}$ ESSCs de la zona inferior correlan la señal de entrada $r[\tau]$ con las secuencias $\{s_{i,j}; 0 \le i \le \frac{M}{2}; 0 \le j \le M-1\}$ que componen $e_{j,1}$. De este modo, a la salida del primer paso se tiene la correlación con las M^2 secuencias complementarias de la matriz $\Delta^{(M,M\cdot L_0}$. En el paso 2 las correlaciones obtenidas se retardan en orden inverso a como se concatenaron las secuencias $s_{i,j}$ en la formación de los pares T-ZCZ$_{2'}$. Así, los resultados de correlación $C_{r,S_{\frac{M}{2}-1}}$ y $C_{r,S_{M-1}}$ no se retardan, a $C_{r,S_{\frac{M}{2}-2}}$ y $C_{r,S_{M-2}}$ se les aplica un retardo igual a la longitud L_0 de los CSS, mientras que a C_{r,S_0} y $C_{r,S_{\frac{M}{2}}}$ el retardo de mayor valor $(\frac{M}{2}-1)\cdot L_0$; en definitiva, $C_{r,s'_{i,j}} = C_{r,s_{i,j}}[\tau-(\frac{M}{2}-1-i)L_0]$ para $0 \le i \le \frac{M}{2}-1$, y $C_{r,s'_{i,j}} = C_{r,s_{i,j}}[\tau-(M-1-i)L_0]$ para $\frac{M}{2} \le i \le M-1$. Además, estos retardos irán multiplicados por $N_{SM} \cdot O_f$ si se consideran los efectos de la modulación y sobremuestreo. Por último, los M sumadores de la parte superior permiten obtener las correlaciones con las secuencias $e_{j,0}$, esto es, $\{C_{r,e_{j,0}} = \sum_{i=0}^{\frac{M}{2}-1} C_{r,s'_{i,j}}; 0 \le j \le M-1\}$; y los M sumadores de la parte inferior la correlación con las secuencias $e_{j,1}$ según $\{C_{r,e_{j,1}} = \sum_{i=\frac{M}{2}}^{M-1} s'_{i,j}; 0 \le j \le M-1\}$.

[2] En la implementación práctica de (5.13) los retardos a aplicar a $C_{r,s_{i,j}}$ aparecen en orden inverso.

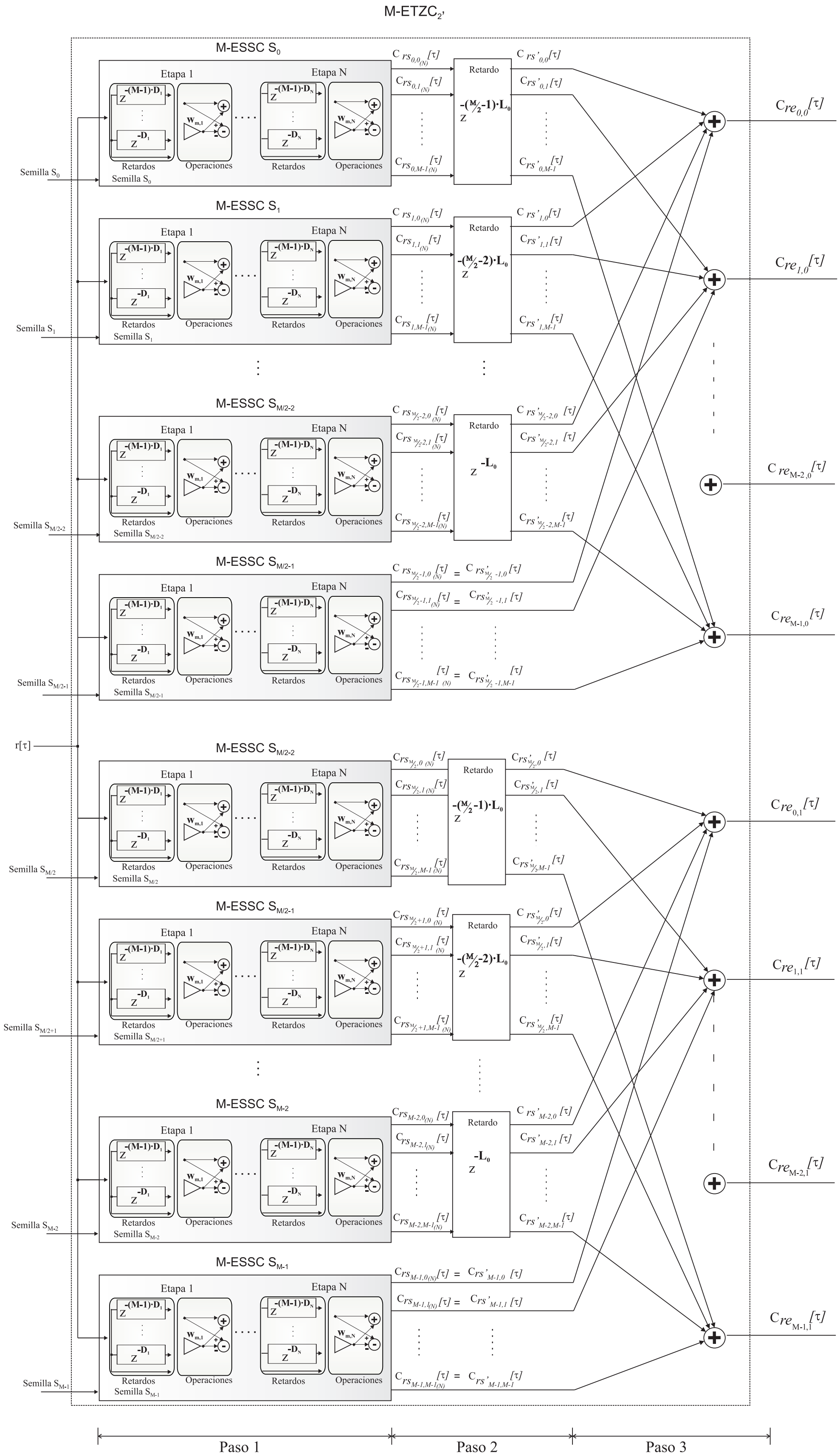


Figura 5.22: Diagrama de bloques del correlador eficiente propuesto para códigos T-ZCZ$_{2'}$ generados con el método 1 a partir de M UCSS (ETZC$_{2'}$).

En la tabla 5.15 se comparan las necesidades computacionales del correlador ETZC$_{2'}$ propuesto con uno directo convencional. Según se ha comentado anteriormente, al trabajar con códigos binarios $\in \{-1, 1\}$ los productos ejecutados en ambos correladores pueden reducirse a sumas y restas. En el caso concreto del ETZC$_{2'}$, estos productos son los efectuados por los ESSCs en el paso 1. También en el paso 1 se realizan un total de M^2mN sumas, mientras que el resto se llevan a cabo en el paso 3. Por otro lado, en la expresión que recoge la totalidad de bits de memoria necesarios en la implementación del ETZC$_{2'}$, el primer sumando corresponde a los bits requeridos por el paso 1, y el segundo a los utilizados en el paso 2. Las necesidades computacionales del correlador directo y el ETZC$_{2'}$ quedan reflejadas en la figura 5.23, cuando se trabaja con una familia de $M = 4$ pares T-ZCZ$_{2'}$.

Implementación	Productos	Sumas	Bits de Memoria
Correlador directo serie	M^{N+2}	$2M(\frac{M^{N+1}}{2} - 1)$	$N_{SM} \cdot O_f \cdot DW \cdot \frac{M^{N+1}}{2} + M^{N+2} + 2M \cdot (DW + mN + m - 1)$
M-ETZC$_{2'}$	$\frac{M^2}{2}mN$	$M^2mN + 2(\frac{M^2}{2} - M)$	$N_{SM} \cdot O_f \cdot DW \cdot (M-1)M^N + N_{SM} \cdot O_f \cdot \frac{(M^3-M^2)}{2}\left[DW\frac{M^{N-1}-1}{M-1} + mM^N\sum_{i=1}^{N-1}\frac{i}{M^{i+1}}\right] + N_{SM} \cdot O_f \cdot (DW + mN) \cdot \frac{M}{2}(\frac{M}{2} - 1) \cdot M^N$

Tabla 5.15: Comparativa de las necesidades computacionales de un correlador directo serie y el ETZC$_{2'}$ propuesto, para la correlación de la señal de entrada con las secuencias de los M pares de una familia T-ZCZ$_{2'}$ de longitud $L = \frac{M^{N+1}}{2}$.

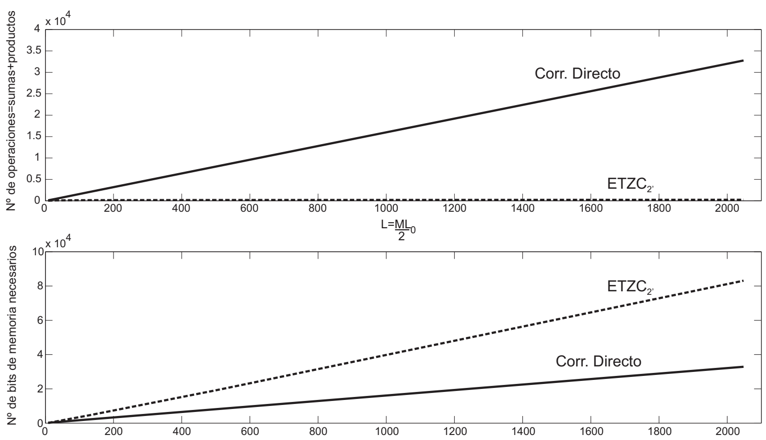


Figura 5.23: Comparativa del número de operaciones y recursos de memoria requeridos por un correlador directo serie y el ETZC$_{2'}$ propuesto, para la detección de $M = 4$ usuarios simultáneos con $DW = 8$.

5.3.4. Estudio de códigos T-ZCZ$_2$ obtenidos a partir de $\Delta^{(M,L_0}$

Utilizando un único CSS también pueden conseguirse pares con tres zonas de correlación cero (T-ZCZ$_2$). Sin embargo, el tamaño de las IFWs es menor que el obtenido empleando los códigos T-ZCZ de la propuesta de Chao Zhang et al. [ZLH04], y tampoco se obtiene una mejora en términos de cota de correlación θ. La ecuación (5.14) especifica el tamaño de las IFWs ($W_A = W_C = E_A = E_C = X_{(N)}$) de pares T-ZCZ$_2$ para una familia de M códigos, en la iteración $n = N$, en donde N es el número de etapas utilizadas en la generación del CSS original.

$$\begin{aligned} n = 1 \quad &\rightarrow \quad X_{(1)} = 0 \\ n = 2 \quad &\rightarrow \quad X_{(2)} = 1 \\ n \geq 3 \quad &\rightarrow \quad X_{(n)} = \tfrac{3M}{2} \cdot M^{n-3} + X_{(n-3)} \end{aligned} \tag{5.14}$$

Por otro lado, el empleo de un único correlador eficiente ESSC para la detección de los M pares no es una solución adecuada. Recuérdese que en este caso un par E_j se forma a partir de una única secuencia complementaria $s_{i,j}$ de un CSS cualquiera S_i, de modo que $\{e_{j,0} = s_{i,j_L} = s_{i,j}[l]; 0 \leq l \leq \frac{L_0}{2} - 1\}$ y $\{e_{j,1} = s_{i,j_R} = s_{i,j}[l]; \frac{L_0}{2} \leq l \leq L_0 - 1\}$. Utilizando un ESSC, en la salida j-ésima del mismo se obtiene $C_{r,s_{i,j}}[\tau]$, esto es la correlación de la señal de entrada $r[\tau]$ con la secuencia $s_{i,j}$ completa, en lugar de la correlación con la mitad de la misma. Por ejemplo, si $r[\tau] = e_{j,0}[\tau]$, entonces $C_{e_{j,0}s_{i,j}}[\tau] = C_{s_{i,j_L},s_{i,j_L}}[\tau] + C_{s_{i,j_L},s_{i,j_R}}[\tau - \frac{L_0}{2}]$; en donde el primer sumando $C_{s_{i,j_L},s_{i,j_L}}$ es la correlación buscada, y el segundo $C_{s_{i,j_L},s_{i,j_R}}$ introduce interferencias no deseadas. Como solución, puede subdividirse el conjunto $S_i^{(M,L_0}$, que da lugar a la matriz $\Delta^{(M,L_0}$ a partir de la que se forman los pares T-ZCZ$_2$, en M subconjuntos $S_i^{(M,\frac{L_0}{M}}$ y realizar la correlación según se ha descrito en 5.3.3. Ahora bien, se necesita conocer de antemano qué subconjuntos $S_i^{(M,\frac{L_0}{M}}$ componen el conjunto $S_i^{(M,L_0}$ y con qué signo han sido concatenados, multiplicando por -1 las salidas de los ESSCs correspondientes.

En definitiva, los códigos T-ZCZ$_{2'}$ son una mejor solución que los T-ZCZ$_2$: las interferencias están acotadas a una zona de menor tamaño y la implementación del correlador requiere menos operaciones.

5.4. Caracterización del método 3 usando $\Delta^{(M,M\cdot L_0}$

5.4.1. Estudio de las propiedades de correlación

El estudio de los códigos T-ZCZ finaliza con el análisis de los obtenidos mediante el método 3, y más específicamente, de los T-ZCZ$_{3'}$ ya que proporcionan una implementación más eficiente del correlador asociado que los T-ZCZ$_3$. Recuérdese que estos códigos T-ZCZ$_{3'}$ tienen longitud $\frac{M^{N+1}}{4}$, siendo $E_j = (e_{j,0}, e_{j,1}) =$

$([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{4}-1,j}],[s_{\frac{M}{4},j}|s_{\frac{M}{4}+1,j}|\cdots|s_{\frac{M}{2}-1,j}])$ y $M>2$ (es decir, en este caso particular, $M=2^m$ con $m\in\mathbb{N}-\{0,1\}$).

Al igual que en los códigos T-ZCZ$_1$, T-ZCZ$_{1'}$, T-ZCZ$_2$ y T-ZCZ$_{2'}$ descritos anteriormente, se obtienen los mismos valores máximos de cota θ y IFWs del mismo tamaño con todas las semillas posibles de los CSSs originales. Por eso, en las tablas 5.16, 5.17 y 5.18, que seleccionan los μ pares de menor cota, se ha omitido el valor de dichas semillas. Obsérvese en dichas tablas que los valores de cota $\theta\leq 0.5$ en todos los casos. Considerando una familia con M códigos, la cota máxima de correlación disminuye conforme aumenta la longitud del código; y como era de esperar, para una misma longitud L la disminución del número de usuarios μ simultáneos permite elegir pares con menores cotas θ. En cuanto a las zonas libres de interferencias, tanto las centrales como las laterales tienen el mismo tamaño $W_A=W_C=E_A=E_C=X_{(N)}$, en donde $X_{(N)}$ se obtiene tras $n=N$ iteraciones según (5.15).

$$\begin{array}{lll} n=1 & \rightarrow & X_{(1)}=\frac{M}{4} \\ n=par & \rightarrow & X_{(n \bmod 2=0)}=M\cdot X_{(n-1)}+2 \\ n=impar & \rightarrow & X_{(n \bmod 2=1)}=M\cdot X_{(n-1)}+\frac{M}{4} \end{array} \tag{5.15}$$

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\%\frac{Int.\ IW_{ACF}}{IW}$	$\%\frac{Int.\ IW_{CCF}}{IW}$
4	2	0.5	0.5	0	0, 2	1	1	100 %	0 %
	3	0.5	0.5	0.5	0, 1, 2				100 %
	4	0.5	0.5	0.5	0, 1, 2, 3				100 %
16	2	0.25	0.25	0	0, 3	6	6	66.67 %	0 %
	3	0.5	0.25	0.5	0, 1, 2				100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
64	2	0.375	0.375	0	0, 2	25	25	30.77 %	0 %
	3	0.375	0.375	0.375	0, 1, 2				46.15 %
	4	0.375	0.375	0.375	0, 1, 2, 3				46.15 %
256	2	0.1875	0.1875	0	0, 1	102	102	23.53 %	0 %
	3	0.375	0.1875	0.375	0, 1, 2				31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %
1024	2	0.2188	0.2188	0	0, 2	409	409	11.71 %	0 %
	3	0.2188	0.2188	0.2188	0, 1, 2				17.56 %
	4	0.2188	0.2188	0.2188	0, 1, 2, 3				17.56 %
4096	2	0.1094	0.1094	0	0, 3	1638	1638	8.79 %	0 %
	3	0.2188	0.1094	0.2188	0, 1, 2				11.72 %
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %

Tabla 5.16: Estudio de la zona con interferencias de familias T-ZCZ$_{3'}$ con $M=4$ códigos.

$\mathbf{M = 8}$

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
16	2	0.5	0.5	0	0, 2	2	2	63.64 %	0 %
	3	0.5	0.5	0.25	0, 2, 5				54.55 %
	4	0.5	0.5	0.25	0, 2, 5, 7				54.55 %
	5	0.5	0.5	0.5	0, 1, 2, 3, 4				81.82 %
	6	0.5	0.5	0.5	0, 1, 2, 3, 4, 5				81.82 %
	7	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6				81.82 %
	8	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7				81.82 %
128	2	0.375	0.375	0	0, 3	18	18	24.18 %	0 %
	3	0.375	0.375	0.1875	0, 2, 5				39.56 %
	4	0.375	0.375	0.1875	0, 2, 5, 7				39.56 %
	5	0.5	0.375	0.5	0, 1, 2, 3, 4				39.56 %
	6	0.5	0.375	0.5	0, 1, 2, 3, 4, 5				39.56 %
	7	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6				39.56 %
	8	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6, 7				39.56 %
1024	2	0.375	0.375	0	0, 3	146	146	15.32 %	0 %
	3	0.375	0.375	0.1406	0, 2, 5				26.67 %
	4	0.375	0.375	0.1406	0, 2, 5, 7				26.67 %
	5	0.375	0.375	0.375	0, 1, 2, 3, 4				26.67 %
	6	0.375	0.375	0.375	0, 1, 2, 3, 4, 5				26.67 %
	7	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6				26.67 %
	8	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7				26.67 %
8192	2	0.2188	0.2188	0	3, 4	1170	1170	10.46 %	0 %
	3	0.2188	0.2188	0.082	1, 3, 4				19.69 %
	4	0.2188	0.2188	0.082	1, 2, 4, 7				19.69 %
	5	0.375	0.2188	0.375	0, 1, 2, 3, 4				19.69 %
	6	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5				19.69 %
	7	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5, 6				19.69 %
	8	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5, 6, 7				19.69 %

Tabla 5.17: Estudio de la zona con interferencias de familias T-ZCZ$_{3'}$ con $M = 8$ códigos.

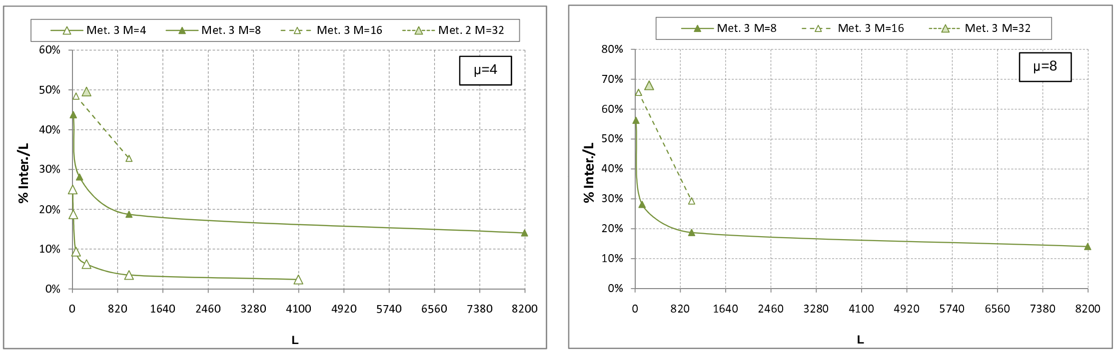
M = 16

L	μ	θ	θ_{AC}	θ_{CC}	códigos	E_A	W_A	$\% \frac{Int.\ IW_{ACF}}{IW}$	$\% \frac{Int.\ IW_{CCF}}{IW}$
64	2	0.5	0.5	0	0, 2	4	4	56.36 %	0 %
	3	0.5	0.5	0.125	0, 2, 13				50.91 %
	4	0.5	0.5	0.125	0, 2, 13, 15				50.91 %
	5	0.5	0.5	0.125	0, 2, 5, 7, 9				76.36 %
	6	0.5	0.5	0.25	0, 2, 5, 7, 9, 11				76.36 %
	7	0.5	0.5	0.25	0, 2, 5, 7, 9, 11, 12				76.36 %
	8	0.5	0.5	0.25	0, 2, 5, 7, 9, 11, 12, 14				76.36 %
	10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				76.36 %
	14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				76.36 %
	16	0.5	0.5	0.5	0, 1, 2, $\cdots$, 15				76.36 %
1024	2	0.375	0.375	0	0, 4	66	66	21.10 %	0 %
	3	0.375	0.375	0.1016	0, 4, 11				37.71 %
	4	0.375	0.375	0.1016	0, 4, 11, 15				37.71 %
	5	0.375	0.375	0.2031	0, 3, 4, 7, 9				33.67 %
	6	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10				33.67 %
	7	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10, 13				33.67 %
	8	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10, 13, 14				33.67 %
	10	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				37.71 %
	14	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				37.71 %
	16	0.5	0.375	0.5	0, 1, 2, $\cdots$, 15				37.71 %

Tabla 5.18: Estudio de la zona con interferencias de familias T-ZCZ$_{3'}$ con $M = 16$ códigos.

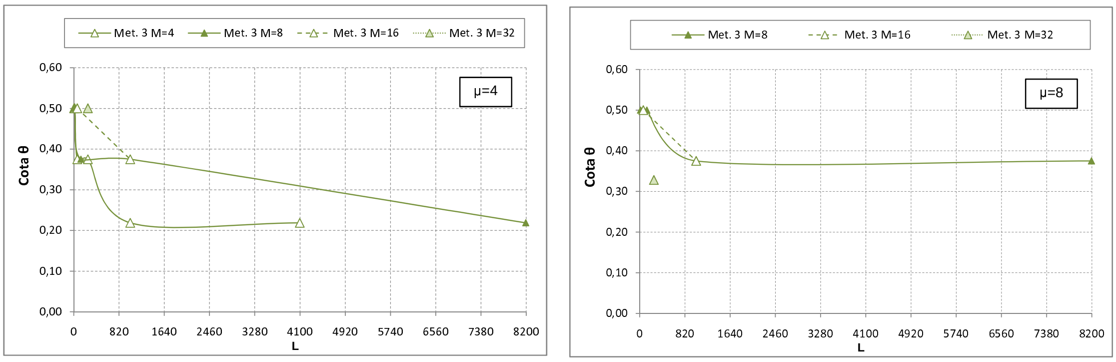
Las figuras de correlación obtenidas con estos pares T-ZCZ$_{3'}$ no distan demasiado de las conseguidas con los métodos 1 y 2: en las zonas con interferencias IW aparecen también valores nulos, y $\theta \leq 0.5$. En la figura 5.24 puede observarse el porcentaje de interferencias en las funciones de correlación, y en las figuras 5.25 y 5.26 los valores máximos de cota de correlación θ en función de la longitud L del código y número $\mu \leq M$ de usuarios simultáneos. Tanto el porcentaje de interferencias como la cota de correlación disminuyen si aumenta la longitud del código y, para una longitud concreta, aumentar el número de usuarios implica un mayor número de interferencias. Para $M = 4$ el número de interferencias y los valores de cota son similares a los obtenidos con códigos T-ZCZ$_1$ de la misma longitud. Si $M > 4$ no hay códigos T-ZCZ$_{3'}$, T-ZCZ$_{2'}$ y T-ZCZ$_1$ de la misma longitud. En cualquier caso, los valores de cota son superiores que los que proporciona el método 1 (T-ZCZ$_1$)

trabajando con códigos de menor longitud. Por otro lado, el porcentaje de interferencias en las funciones de correlación es parecido al obtenido con pares T-ZCZ$_{2'}$ del doble de longitud.



(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.24: Porcentaje de interferencias en pares T-ZCZ$_{3'}$ en función del número M de códigos de la familia y la longitud L de los mismos para $\mu = 4$ y $\mu = 8$ usuarios simultáneos.



(a) $\mu = 4$ (b) $\mu = 8$

Figura 5.25: Valores de cota en la IW de pares T-ZCZ$_{3'}$ en función del número M de códigos disponibles en la familia y de la longitud L de los mismos.

Utilizando el método 3 propuesto por Chao Zhang et al. [ZLH04] los valores de cota se mantienen constantes en $\theta = 0.5$ independientemente de los valores de M, L y μ. Con los pares T-ZCZ$_{3'}$, sin embargo, es posible conseguir $\theta < 0.5$ eligiendo adecuadamente los parámetros anteriores, tal y como puede observarse en la figura 5.27.

5.4.2. Influencia de la recepción parcial del código T-ZCZ$_{3'}$ en la cota de auto-correlación

En la figura 5.28 se muestra el valor de la cota de auto-correlación θ_{AC} en la IW (línea continua) y en la IFW (línea discontinua) en función del porcentaje de código recibido. Los valores representados corresponden a las cotas θ_{AC} máximas obtenidas con los $\mu = 4$ usuarios especificados en las tablas 5.16 a 5.18, para códigos de longitud $L = 64$ bits

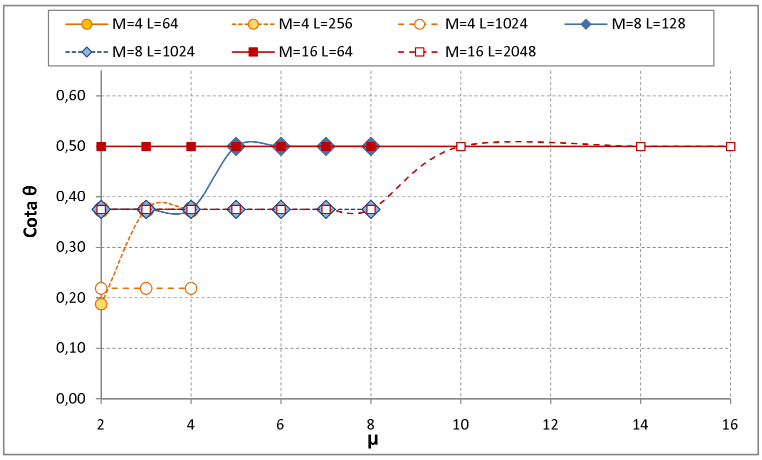


Figura 5.26: Valores de cota θ en la IW de pares T-ZCZ$_{3'}$ en función del número M de códigos disponibles en la familia, su longitud L y el número μ de usuarios simultáneos.

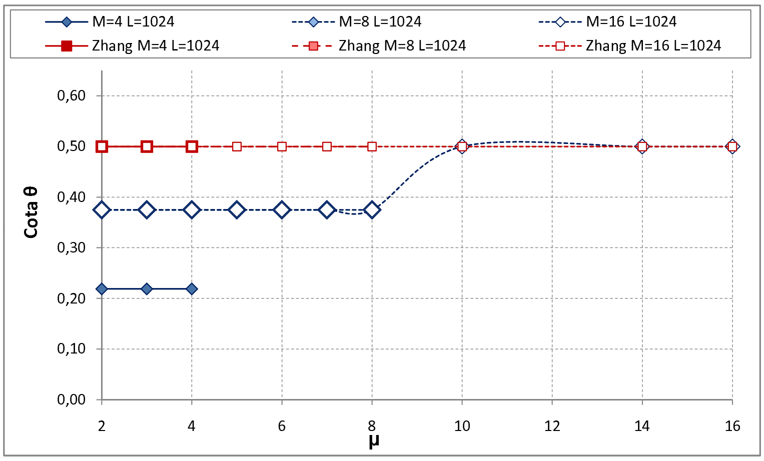
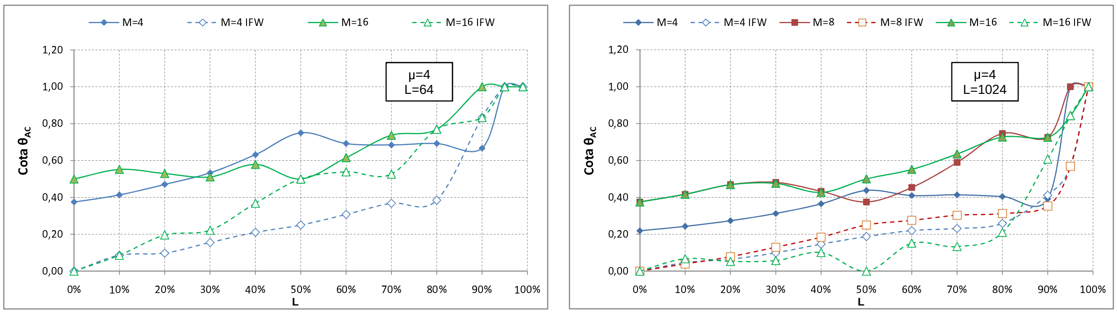


Figura 5.27: Comparativa de los valores de cota θ máximos de pares T-ZCZ$_{3'}$ obtenidos con el método 3 propuesto y el método 3 de Chao Zhang [ZLH04].

y $L = 1024$ bits. Se obtienen resultados similares con un mayor número de usuarios simultáneos.



(a) $L = 64$ (b) $L = 1024$

Figura 5.28: θ_{AC} en función del porcentaje de código perdido para $\mu = 4$ códigos T-ZCZ$_{3'}$.

5.4.3. Propuesta de un correlador eficiente

Las bases sobre las que se fundamenta el correlador eficiente de códigos T-ZCZ$_{3'}$ propuesto (ETZC$_{3'}$) y las de los correladores ETZC$_{1'}$ y ETZC$_{2'}$ son las mismas; teniendo en cuenta ahora que una secuencia del par T-ZCZ$_{3'}$ contiene $\frac{M}{4}$ secuencias concatenadas, cada una perteneciente a un conjunto incorrelado distinto. De este modo, la correlación de una señal $r[\tau]$ con las secuencias $(e_{j,0}, e_{j,1})$ de un par T-ZCZ$_{3'}$ puede descomponerse en la correlación de dicha señal de entrada $r[\tau]$ con las secuencias $s_{i,j}$ que componen el par, según (5.16) ó (5.17) dependiendo de si el conjunto E_j se generó a partir de la mitad izquierda $\Delta_L^{(M,M\cdot L_0}$ de la matriz $\Delta^{(M,M\cdot L_0}$ o de la derecha $\Delta_R^{(M,M\cdot L_0}$. La estructura del correlador ETZC$_{3'}$ es la mostrada en la figura 5.29 para códigos T-ZCZ$_{3'}$ obtenidos a partir de $\Delta_L^{(M,M\cdot L_0}$; sustituyendo las semillas de los ESSCs por las correspondientes a los conjuntos $\{S_i; \frac{M}{2} \leq i \leq M-1\}$ se obtiene el correlador asociado a códigos T-ZCZ$_{3'}$ generados a partir de $\Delta_R^{(M,M\cdot L_0}$. A partir del ETZC$_{3'}$ aquí propuesto puede derivarse un generador eficiente ETZG$_{3'}$ con solo permutar el orden de las etapas de retardo [Pop99].

$$\begin{aligned} C_{r,e_{j,0}} &= C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{4}-1,j}}[\tau - (\tfrac{M}{4}-1)L_0] \\ C_{r,e_{j,1}} &= C_{r,s_{\frac{M}{4},j}}[\tau] + C_{r,s_{\frac{M}{4}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau - (\tfrac{M}{4}-1)L_0] \end{aligned} \tag{5.16}$$

$$\begin{aligned} C_{r,e_{j,0}} &= C_{r,s_{\frac{M}{2},j}}[\tau] + C_{r,s_{\frac{M}{2}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{3M}{4}-1,j}}[\tau - (\tfrac{M}{4}-1)L_0] \\ C_{r,e_{j,1}} &= C_{r,s_{\frac{3M}{4},j}}[\tau] + C_{r,s_{\frac{3M}{4}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{M-1,j}}[\tau - (\tfrac{M}{4}-1)L_0] \end{aligned} \tag{5.17}$$

Las bondades de este correlador frente a uno directo quedan reflejadas en la tabla 5.19. El ETZC$_{3'}$ realiza la correlación de una señal $r[\tau]$ con las secuencias de los M pares de una

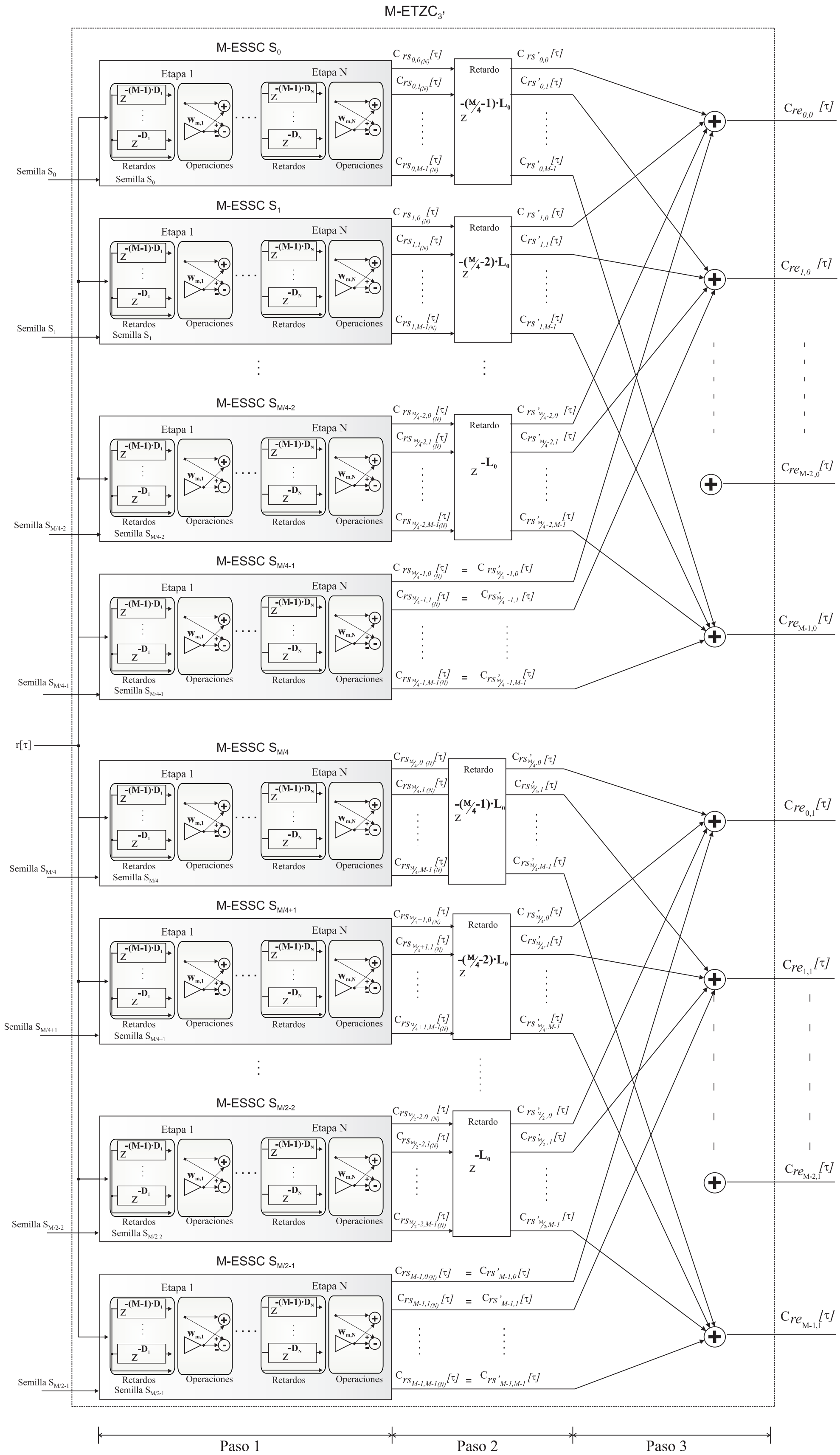


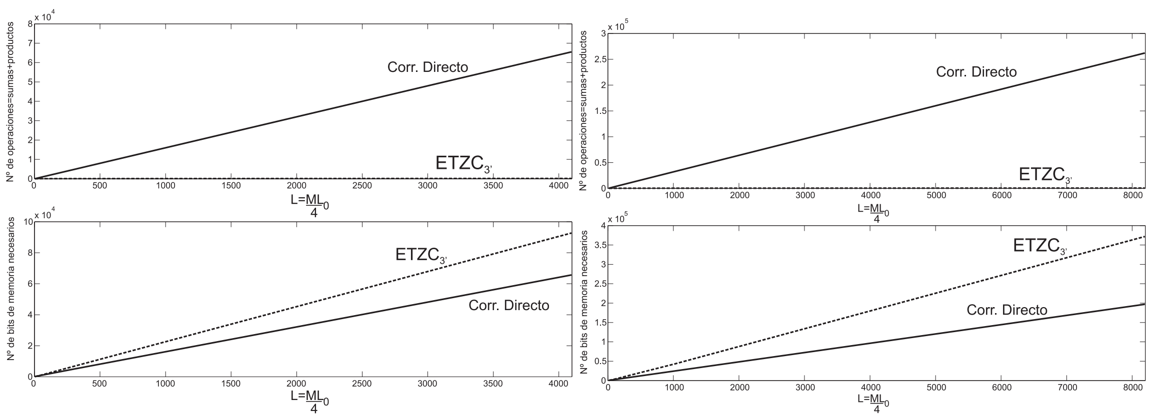
Figura 5.29: Diagrama de bloques del correlador eficiente propuesto para códigos T-ZCZ$_{3'}$ generados con el método 1 a partir de M UCSS (ETZC$_{3'}$).

familia T-ZCZ$_{3'}$ empleando menos operaciones que las que necesita uno directo; ahora bien, requiere almacenar en memoria un mayor número de bits. Para una mejor comparación se ha representado en la figura 5.30 las necesidades computacionales de ambos correladores cuando se trabaja con familias con $M = 4$ y $M = 8$ códigos T-ZCZ$_{3'}$.

Como en las plataformas configurables actuales, y para las longitudes y número de usuarios empleados, la restricción se encuentra en el tiempo de correlación y no en la capacidad de almacenamiento, puede concluirse que el ETZC$_{3'}$ es más adecuado.

Implementación	Productos	Sumas	Bits de Memoria
Correlador directo serie	$\frac{M^{N+2}}{2}$	$2M(\frac{M^{N+1}}{4} - 1)$	$N_{SM} \cdot O_f \cdot DW \cdot \frac{M^{N+1}}{4} + \frac{M^{N+2}}{2} + 2M \cdot (DW + mN + m - 2)$
M-ETZC$_{3'}$	$\frac{M^2}{4} mN$	$\frac{M^2}{2} mN + 2(\frac{M^2}{4} - M)$	$N_{SM} \cdot O_f \cdot DW \cdot (M - 1)\frac{M^N}{2} + N_{SM} \cdot O_f \cdot \frac{(M^3-M^2)}{4}\left[DW\frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right] + N_{SM} \cdot O_f \cdot (DW + mN) \cdot (\frac{M^2}{16} - \frac{M}{4})M^N$

Tabla 5.19: Comparativa de las necesidades computacionales de un correlador directo serie y el ETZC$_{3'}$ propuesto, para la correlación de la señal de entrada con las secuencias de los M pares de una familia T-ZCZ$_{3'}$ de longitud $L = \frac{M^{N+1}}{4}$.



(a) $M = 4$ (b) $M = 8$

Figura 5.30: θ_{AC} en función del porcentaje de código perdido para $\mu = 4$ códigos T-ZCZ$_{3'}$.

5.4.4. Códigos T-ZCZ$_3$ obtenidos a partir de $\Delta^{(M,L_0}$

El empleo de códigos T-ZCZ$_3$ generados a partir de un único CSS $S_i^{(M,L_0}$ se ha descartado por razones similares a las esgrimidas para no emplear pares T-ZCZ$_2$. Esto es, el área de interferencias IW es mayor en estos códigos que en los propuestos en [ZLH04], y utilizar un único ESSC implica la aparición de interferencias en las funciones de correlación. Podría utilizarse una estructura similar a la de la figura 5.29 si el conjunto $S_i^{(M,L_0}$ se descompone