

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**SIMULACIÓN DE ESTIMACIÓN DE CANAL Y SINCRONIZACIÓN
EN LA TÉCNICA DE COMUNICACIÓN INALÁMBRICA SEFDM
SIMULACIÓN DE STO (SYMBOL TIME OFFSET) EN SEFDM**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
TELECOMUNICACIONES**

JORGE EDUARDO CASTILLO CARPIO

jorge.castillo@epn.edu.ec

DIRECTOR: DIEGO JAVIER REINOSO CHISAGUANO

diego.reinoso@epn.edu.ec

DMQ, octubre 2022

CERTIFICACIONES

Yo, JORGE EDUARDO CASTILLO CARPIO declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



JORGE EDUARDO CASTILLO CARPIO

Certifico que el presente trabajo de integración curricular fue desarrollado por JORGE EDUARDO CASTILLO CARPIO, bajo mi supervisión.



DIEGO JAVIER REINOSO CHISAGUANO
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

JORGE EDUARDO CASTILLO CARPIO

DIEGO JAVIER REINOSO CHISAGUANO

DEDICATORIA

A Dios en primer lugar que me ha brindado sabiduría e inteligencia, a lo largo de mi carrera nunca me dejó solo y me ayudó a superar todo obstáculo.

A mis padres Rodrigo, Luis y Jimena quienes me guiaron por el camino correcto y me brindaron su apoyo incondicional, por formarme como un hombre de bien para así poder cumplir de la mejor manera con mis metas.

A mis hermanas Alexandra y Verónica quienes me reconfortaron en cada momento, siendo para ellas el mejor ejemplo de valentía y responsabilidad.

AGRADECIMIENTO

Quiero agradecer a Dios por permitirme adquirir nuevos conocimientos, por brindarme sabiduría para tomar buenas decisiones y por darme la oportunidad de estudiar.

A mi familia por ser el motor de mi vida que me impulso a seguir adelante, a mi enamorada Genesis por estar siempre apoyándome para no rendirme en momentos difíciles.

A mis amigos Ricardo y Danny que compartieron momentos amenos y fueron una fuente de motivación en cada conversación.

Al Dr. Diego Reinoso que ha sabido guiarme durante el proyecto con paciencia y sabiduría para poder avanzar en cada una de las etapas de desarrollo.

ÍNDICE DE CONTENIDO

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN.....	VII
ABSTRACT.....	VIII
1 INTRODUCCIÓN.....	1
1.1 OBJETIVO GENERAL	1
1.2 OBJETIVOS ESPECÍFICOS	1
1.4 MARCO TEÓRICO	2
1.4.1 SISTEMA SEFDM	2
1.4.2 SINCRONIZACIÓN EN TIEMPO	6
2 METODOLOGÍA.....	11
2.1 PROGRAMA PRINCIPAL	11
2.2 SISTEMA DE TRANSMISIÓN SEFDM	13
2.2.1 VALORES INICIALES	14
2.2.2 GENERACIÓN DE BITS	15
2.2.3 CODIFICACIÓN.....	16
2.2.4 MODULACIÓN	16
2.2.5 IFFT	16
2.2.6 INSERCIÓN DEL PREFIJO CÍCLICO	17
2.2.7 PREÁMBULO	17
2.3 CANAL AWGN.....	20
2.4 SISTEMA DE RECEPCIÓN SEFDM.....	20
2.4.1 SINCRONIZACIÓN SEFDM.....	22
2.4.2 RETIRAR PREÁMBULO	32
2.4.3 RETIRAR PREFIJO CÍCLICO / FFT	32
2.4.4 DETECCIÓN.....	33
2.4.5 DEMODULACIÓN	34
2.4.6 DECODIFICACIÓN	34
2.4.7 CÁLCULO DE BITS ERRADOS	35

3	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES	36
3.1	ESCENARIO DE SIMULACIÓN	36
3.2	PRUEBAS DEL SISTEMA	36
3.3	RESULTADOS.....	37
3.4	CONCLUSIONES	44
3.5	RECOMENDACIONES	46
4	REFERENCIAS BIBLIOGRÁFICAS.....	47
5	ANEXOS	48
	ANEXO I	48

RESUMEN

En este trabajo de integración curricular se presenta la simulación de un sistema SEFDM (*Spectrally Efficient Frequency Division Multiplexing*) donde se realiza sincronización en el dominio del tiempo y principalmente se realiza estimación de STO (*Symbol Time Offset*). En el receptor del sistema se emplean tres técnicas de estimación STO donde se utiliza el prefijo cíclico y el preámbulo con la finalidad de determinar el inicio del símbolo. El sistema SEFDM permite comprimir el ancho de banda agregando ceros a la señal y en su etapa de sincronización es importante reconocer como el efecto STO se agrega, se estima, y se compensa.

En el primer capítulo se realiza una revisión teórica del sistema SEFDM y de la sincronización en tiempo. Se revisa también las técnicas de estimación STO que se van a implementar para el sistema mencionado.

En el segundo capítulo, se describe el código desarrollado en MATLAB, y se explica por medio de diagramas de bloques el transmisor y receptor. En el transmisor se realiza especial énfasis en la construcción del preámbulo, para el receptor se realiza hincapié en la etapa de sincronización.

Para concluir con el trabajo se tiene el tercer capítulo donde se presentan los resultados con su correspondiente análisis de los escenarios de simulación más relevantes, para lo cual se emplean los diferentes tipos de modulación y valores específicos de factor de compresión de ancho de banda. Estos resultados se presentan por medio de gráficas del BER (*Bit Error Rate*) y E_b/N_0 (*Energy per-bit to noise power density ratio*). Finalmente, se incluyen las conclusiones y se añade recomendaciones para posibles futuros trabajos.

PALABRAS CLAVE: STO, SEFDM, factor de compresión, sincronización.

ABSTRACT

This work presents the simulation of a SEFDM (Spectrally Efficient Frequency Division Multiplexing) system where synchronization is performed in the time domain and mainly STO (Symbol Time Offset) estimation is performed. Three STO estimation techniques are used in the system receiver where the cyclic prefix and the preamble are used to determine the beginning of the symbol. The SEFDM system allows compressing the bandwidth by adding zeros to the signal and in its synchronization stage is important to recognize how the STO effect is added, estimated, and compensated.

The first chapter provides a theoretical review of the SEFDM system and time synchronization. It also reviews the STO estimation techniques to be implemented for the system.

In the second chapter, the code developed in MATLAB is described and the transmitter and receiver are explained by means of block diagrams. In the transmitter, special emphasis is made on the construction of the preamble, for the receiver the emphasis is made on the synchronization stage.

To conclude the work, the third chapter presents the results with their corresponding analysis of the most relevant simulation scenarios, for which the different types of modulation and specific values of bandwidth compression factor are used. These results are presented with graphs of the BER (Bit Error Rate) and E_b/N_0 (Energy per-bit to noise power density ratio). Finally, the conclusions are presented along with recommendations for possible future work.

KEYWORDS: STO, SEFDM, compression ratio, synchronization.

1 INTRODUCCIÓN

Las comunicaciones inalámbricas surgieron a medida que avanzó la comunicación en la humanidad, a medida que se fueron implementando en tecnologías como sistemas celulares o satelitales, debido a la flexibilidad que brinda al implementarla como medio de transmisión [1].

Debido a que las comunicaciones inalámbricas son el medio por el cual los equipos móviles o fijos se comunican intercambiando información y a su vez usando al espectro electromagnético se necesitan técnicas de transmisión multiportadora como OFDM (*Orthogonal frequency division multiplexing*) que se usan en varias aplicaciones inalámbricas con el objetivo de transmitir datos [1]. OFDM se caracteriza por presentar robustez frente al multitrayecto, pero depende de la ortogonalidad de las subportadoras. En cambio, SEFDM ha surgido en base a la necesidad de tener eficiencia espectral es decir reducir el ancho de banda y transmitir la misma cantidad de información, cabe recalcar que su rendimiento ya no depende de la ortogonalidad entre subportadoras [2].

El receptor del sistema SEFDM desconoce donde inicia la señal el momento que recibe los datos y se produce el efecto STO. SEFDM al perder la ortogonalidad requiere un proceso de sincronización en el receptor. Para compensar el efecto del STO se requieren técnicas de estimación de STO por lo cual se van a utilizar técnicas basadas en prefijo cíclico y preámbulo [3].

En el trabajo de integración que se presenta se realiza una simulación e implementación en MATLAB de un sistema SEFDM, el efecto STO aparece en el receptor al recibir la señal ya que se desconoce el inicio del símbolo. Para poder llevar a cabo la sincronización se realiza una revisión de las técnicas de estimación, para así determinar el inicio del símbolo. Además, se usa la tasa de errores y E_b/N_0 como métricas para graficar y determinar el rendimiento de cada técnica en el sistema.

1.1 OBJETIVO GENERAL

- Simular la estimación STO en el receptor de un sistema SEFDM

1.2 OBJETIVOS ESPECÍFICOS

- Realizar una revisión sobre técnicas de estimación de STO.
- Diseñar el diagrama de bloques de la simulación.
- Implementar la simulación en MATLAB.

- Realizar las pruebas de funcionamiento con diferentes parámetros.
- Analizar los resultados obtenidos de la simulación.

1.3 ALCANCE

En la simulación se propone un sistema de comunicaciones inalámbricas con SEFDM que permita realizar sincronización en el receptor. Se realizará una revisión sobre técnicas de estimación de STO. El sistema SEFDM está compuesto de un transmisor y un receptor a través un canal AWGN. Se utilizará la codificación convolucional de tasa $\frac{1}{2}$ es decir que por cada bit que entra salen dos, lo cual servirá para agregar redundancia al sistema. En el receptor se utilizará una o dos técnicas de estimación STO determinando el punto de inicio del símbolo SEFDM para poder llevar a cabo la sincronización en tiempo. La simulación permitirá utilizar modulación QPSK, 16 QAM o 64 QAM y un factor de compresión de ancho de banda configurable. Para recuperar la señal se utilizará la técnica MMSE. Se va a utilizar un tamaño de ventana de 16 para la IFFT (*Inverse Fast Fourier transform*) y FFT (*Fast Fourier transform*). Los resultados que se tendrán al final serán en gráficas de BER en función de Eb/No.

La metodología de investigación que se va a realizar es cuantitativa. A continuación, se va a detallar las fases que posee. En la fase de diseño se genera el diagrama de bloques de la simulación. La fase de implementación consiste en implementar la simulación en MATLAB. Por último, en la fase de análisis de resultados se generan las gráficas para comparar el caso ideal con los casos: estimación STO, sin estimación STO y analizar su rendimiento. Otro análisis sería variando los parámetros de modulación y factor de compresión de ancho de banda. Se analizará los resultados con el fin de obtener conclusiones y recomendaciones.

1.4 MARCO TEÓRICO

Se describe un sistema SEFDM considerando conceptos básicos para la transmisión y recepción en el cual se desea implementar técnicas para contrarrestar los efectos del STO usando técnicas en el dominio del tiempo.

1.4.1 SISTEMA SEFDM

SEFDM es una técnica de acceso multiportadora que se basa en OFDM, se diferencia de OFDM debido a la ausencia de la propiedad de ortogonalidad entre portadoras. Posee una alta eficiencia espectral lo cual quiere decir que permite enviar la misma cantidad de datos

usando un ancho de banda menor, reduciendo el espacio en frecuencia entre subportadoras [4].

Su funcionamiento se basa en ingresar cierta cantidad de datos, se divide en N subportadoras paralelas con menor velocidad comparado con las que ingresaron. La separación entre subportadoras Δf es igual a la fracción entre α (factor de compresión) y T (periodo del símbolo), para $\alpha < 1$. Para entender mejor el factor de compresión se utiliza la Figura 1.1 donde se muestra para $\alpha = 1$ en OFDM, $\alpha = 0.75$ quiere decir $(1 - \alpha = 0.25)$ veinte y cinco por ciento de ahorro en ancho de banda y $\alpha = 0.5$ (50 % de ahorro) [5].

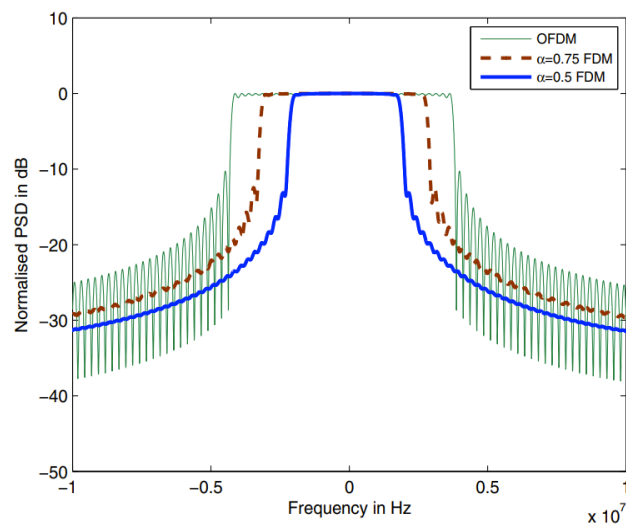
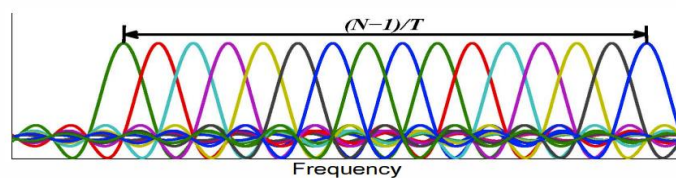
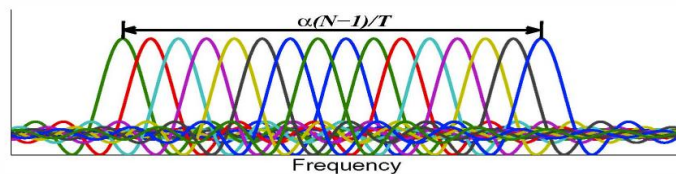


Figura 1.1. Espectro para $N=32$, $T=4(\mu\text{sec})$ con $\alpha = \{1, 0.75, 0.5\}$ [5]



(a) OFDM Spectrum.



(b) SEFDM Spectrum with $\alpha = 0.8$.

Figura 1.2. Espectro de 16 subportadoras [6].

SEFDM es básicamente un esquema FDM con eficiencia espectral, sin embargo, el reducir el espacio entre subportadora conlleva el hecho de que exista ICI (*Inter Interference Carrier*) y por ende degradación del BER. La principal comparación es con OFDM ya que es muy utilizado en el ámbito de las comunicaciones inalámbricas. En la Figura 1.2 se observa lo que implica utilizar el factor de compresión en el espaciado de las subportadoras tanto para OFDM como en SEFDM para un factor de compresión de 0.8 [5].

1.4.1.1 Transmisor y Receptor

A continuación, se revisa los componentes de un sistema de comunicaciones inalámbricas con SEFDM. Su análisis y comprensión resulta de ayuda para la ubicación de la sincronización en tiempo siendo el receptor en donde se van a colocar las técnicas de estimación en tiempo.

a) Conversión serie a paralelo/ paralelo a serie

Los datos ingresan a un convertidor serie/paralelo para luego modularse las N subportadoras, dicha conversión de datos disminuye la velocidad de símbolo de cada subportadora por un factor de N , lo que implica un aumento de N veces en la duración del símbolo. Del flujo en paralelo se aprovecha para la transformada inversa de Fourier. Antes de que atraviesen el canal de propagación se realiza el proceso contrario.

b) Modulación/ demodulación digital

Para la modulación SEFDM se pueden usar las modulaciones como BPSK, QAM pero a medida que se incrementa el nivel de modulación, pueden brindar mayor eficiencia espectral. Pero al conocer que en los diagramas de constelación a medida que se incrementa el nivel la distancia entre vecinos se acorta se considera que a mayor nivel de modulación se vuelven menos resistentes a la interferencia. Básicamente lo que pretende la modulación es aumentar la velocidad con la que se transmiten los datos [7].

c) IFFT/FFT

En el transmisor se realiza la transformada inversa de Fourier es decir se cambian los datos de frecuencia a tiempo, en cambio en el receptor se realiza la transformada directa de Fourier en pocas palabras del dominio del tiempo a frecuencia. En el caso de un sistema SEFDM antes de aplicar la IFFT se insertan ceros al final y después de ese proceso se toma la información sin tomar en cuenta esos valores, sucede lo mismo para la FFT en el receptor [7].

d) Canal

El modelo más simple que se utiliza es el modelo de canal AWGN que se caracteriza por ser estocástico es decir aleatorio, al ser ruido térmico se lo puede tratar como un proceso ergódico y estacionario. Un proceso es ergódico si sus características pueden ser conocidas a partir de una sola observación a lo largo de un periodo de tiempo y estacionario porque sus características estadísticas no varían con el tiempo. El ruido térmico es un proceso estacionario, tiene un valor medio nulo y su potencia es constante [7].

En la figura 1.3 se observa, en forma general un sistema SEFDM:

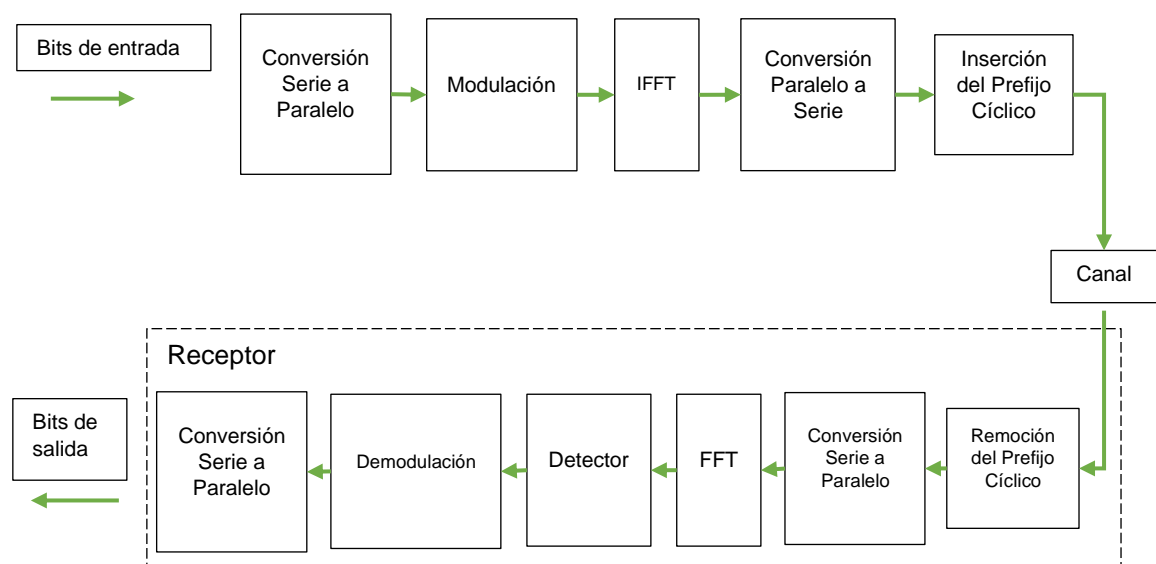


Figura 1.3. Diagrama de bloques del sistema SEFDM [7].

e) Prefijo cíclico

El prefijo cíclico es un conjunto de valores que contiene una réplica de muestras del símbolo, con el propósito de reducir el ISI (*Inter Symbol Interference*) se agregan las muestras al comienzo del símbolo. Además, para reducir el ICI, el prefijo cíclico contiene una extensión cíclica con las últimas muestras del símbolo es decir se toman los valores que contiene el símbolo al final. Es importante tener en cuenta la inserción y remoción del prefijo cíclico para el sistema SEFDM ya que al perder la ortogonalidad se presentan los dos fenómenos [7].

1.4.1.2 Ventajas y Desventajas

Algunas ventajas y desventajas:

- SEFDM provee un sistema con un factor de compresión regulable que permite aumentar la eficiencia espectral.
- Algo importante que cabe mencionar es que SEFDM genera un desafío ya que no se sabe con seguridad si se puede generar un gran número de subportadoras sin ortogonalidad en el mismo ancho de banda ya que podría incurrir en un alto consumo de energía [6].
- Alto factor de PAPR (*Peak to Average Power Ratio*) que lo hace sensible a las no linealidades introducidas por el amplificador de potencia.
- El ancho de banda es un recurso costoso con SEFDM se puede reducir costos, pero al ir reduciendo el factor de compresión se tiene mayor complejidad en detección.

1.4.2 SINCRONIZACIÓN EN TIEMPO

La sincronización en tiempo resulta ser de vital importancia en las comunicaciones inalámbricas, permite contrarrestar el efecto de adelanto o retraso del tiempo del símbolo. Se revisa a continuación los efectos y las técnicas que se aplican para contrarrestar en OFDM el STO y es preciso recalcar que se las va a aplicar también para un sistema SEFDM.

1.4.2.1 Efecto del STO

Para SEFDM como se observa en la sección anterior posee en su sistema de comunicaciones, para transmisión la IFFT y recepción se realiza la FFT, siendo funciones fundamentales para el posterior análisis. Se requiere contrarrestar el efecto del STO para obtener las muestras exactas del símbolo para el número de puntos de FFT o IFFT, sin incluir el prefijo cíclico. El objetivo es determinar donde comienza el símbolo.

En la tabla 1.1 se observa que el desplazamiento en tiempo o error se denomina δ lo cual provoca un desplazamiento en fase en el dominio de la frecuencia, siendo N el número de subportadoras y k el índice de subportadora. La variable y representa la señal recibida, mientras que x representa señal transmitida [8].

Tabla 1.1. Efecto del STO [8].

	Señal recibida	STO(δ)
Dominio del tiempo	$y[n]$	$x[n + \delta]$
Dominio de la frecuencia	$Y[k]$	$e^{j2\pi k\delta/N} X[k]$

Se tiene cuatro posibles escenarios donde el efecto del STO puede actuar de diferente manera, que se observa en la Figura 1.4. El primer caso es ideal y no existe desplazamiento del símbolo, en el segundo caso empieza unos instantes antes, el siguiente se adelanta demasiado al inicio del símbolo, y finalmente, el símbolo inicia unos instantes después del inicio del símbolo [8].

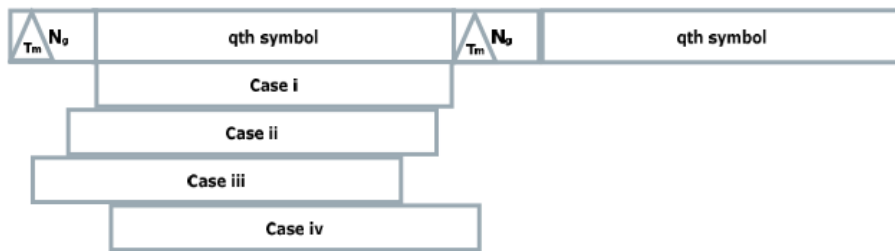


Figura 1.4. Casos de estimación con presencia de STO [9].

1.4.2.2 Técnicas de estimación del STO

Se realiza una revisión de las técnicas para contrarrestar los efectos de STO en OFDM para implementar en un sistema SEFDM en el lado del receptor. Con la finalidad de obtener el inicio exacto del símbolo en el dominio del tiempo se detalla a continuación los tipos estimación. Se va a utilizar la nomenclatura dentro del prefijo cíclico N_G muestras en T_G segundos y dentro de los datos enviados en un símbolo N_{sub} muestras en T_{sub} segundos [8].

Estimación del STO utilizando el prefijo cíclico

El prefijo cíclico sirve como técnica de estimación debido a que la información que contiene se puede utilizar para comparar valores entre símbolos. En la figura 1.5 se observan dos ventanas deslizantes que constan de N_G muestras con T_G de duración, además, se encuentran separadas por N_{sub} muestras correspondientes a un símbolo que se indica en la parte C. La parte B y B' corresponden a las ventanas W1 y W2 respectivamente. Cabe recalcar que tanto el prefijo como las ventanas deslizantes constan de las mismas muestras siendo así bloques idénticos [8]. Para la estimación se deslizan las ventanas con la finalidad

de encontrar semejanza sabiendo que el prefijo cíclico está al inicio del símbolo. Parte clave en la estimación van a ser los intervalos del prefijo cíclico y del símbolo [9].

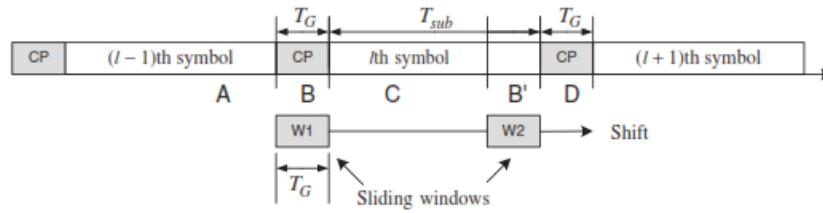


Figura 1.5. Estimación del símbolo utilizando ventanas deslizantes [8].

Para determinar el inicio del símbolo se requiere minimizar la diferencia o maximizar la correlación, mostradas en las siguientes ecuaciones (1.1) y (1.2), respectivamente [9].

$$\delta = \min \left\{ \sum_{j=\delta}^{N_G-1+\delta} |y[n+j] - y[n+N+j]| \right\} \quad (1.1)$$

$$\delta = \max \left\{ \sum_{j=\delta}^{N_G-1+\delta} |y[n+j] * y^*[n+N+j]| \right\} \quad (1.2)$$

La ecuación (1.2) permite estimar la máxima correlación entre los bloques W1 y W2 con la finalidad de cuantificar la similitud. Este tipo de estimación se la conoce como ML (*Maximum-Likelihood Estimation Scheme*).

Estimación del STO utilizando símbolo de entrenamiento

Consiste en enviar dos bloques idénticos o un solo símbolo de entrenamiento con una estructura repetitiva. En la Figura 1.6 se observa que W1 y W2 tienen un periodo $T_{sub}/2$ de las muestras de datos [8].

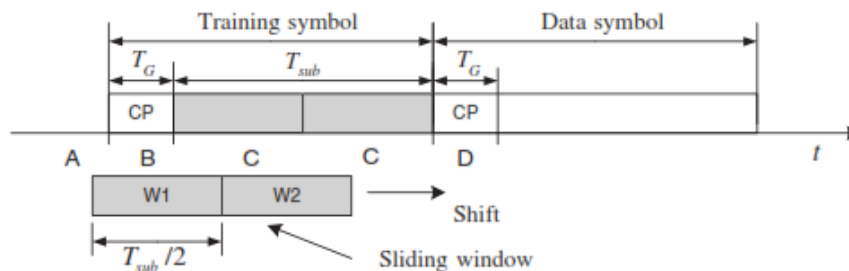


Figura 1.6. Estimación STO utilizando símbolos de entrenamiento (periodo = $T_{sub}/2$) [8].

Existen dos bloques deslizantes, W1 y W2. En este caso se diferencian en las ventanas representadas por C, con duración $T_{sub}/2$, de las ventanas A, B y D de tal forma que se pueda estimar el STO. La ecuación (1.3) permite determinar el punto donde la diferencia sea mínima entre los bloques W1 y W2 [8].

$$\delta = \min \left\{ \sum_{j=\delta}^{\frac{N}{2}-1+\delta} \left| y[n+j] - y\left[n + \frac{N}{2} + j\right] \right| \right\} \quad (1.3)$$

En la figura 1.7 se tienen dos ventanas deslizantes, W1 y W2. Estas ventanas se requieren para diferenciar los bloques representados por C, con duración $T_{sub}/4$, de los bloques A, B y D; de tal forma que se pueda estimar el STO. Las ventanas deslizantes ocupan los espacios representados con C, pero se mantienen los dos bloques [8].

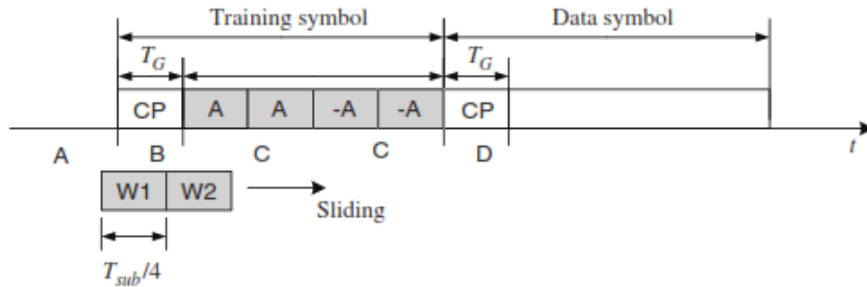


Figura 1.7. Estimación STO utilizando símbolos de entrenamiento (periodo = $T_{sub}/4$)

La ecuación (1.4) permite encontrar el punto donde la diferencia sea mínima entre dos bloques W1 y W2 [8].

$$\delta = \min \left\{ \sum_{j=\delta}^{\frac{N}{4}-1+\delta} \left| y[n+j] - y\left[n + \frac{N}{4} + j\right] \right| \right\} \quad (1.4)$$

La siguiente técnica es similar a la anterior ya que se basa en la diferencia mínima, pero utiliza el preámbulo conocido en el transmisor y receptor. Se realiza el proceso anterior buscando la diferencia mínima solo con la parte del símbolo conformado por A y luego -A es decir el preámbulo se mantiene estático en las comparaciones correspondiente al tamaño de ventana establecida anteriormente. En la Ecuación (1.5) se tiene el punto de diferencia mínima entre W y el preámbulo [8].

$$\delta = \min \left\{ \sum_{j=\delta}^{N_G-1+\delta} |y[n+j] - pream| \right\} \quad (1.5)$$

2 METODOLOGÍA

En este capítulo se presenta a detalle el código implementado para la simulación del sistema SEFDM que utiliza un canal AWGN (Additive white Gaussian noise), a través del software Matlab. La investigación se ha realizado mediante un enfoque cuantitativo, al utilizar técnicas de estimación, el objetivo es observar el rendimiento de las diferentes técnicas considerando lo que se ha investigado previamente del problema. Además, como parámetro de medición se utiliza la cantidad de errores, se delimita el problema y en concreto lo que se desea implementar es la sincronización en tiempo STO en un sistema SEFDM.

Para la implementación del sistema SEFDM se sigue un orden secuencial, es decir necesariamente tiene que realizar cada uno de los pasos de forma seguida. El sistema SEFDM consta de tres partes fundamentales para su funcionamiento que son: transmisor, canal con ruido AWGN y el receptor con su respectiva sincronización [4].

Para su análisis se generan gráficas comparativas que muestran varias curvas, ya que eso permite visualizar y comparar de forma correcta la calidad de las técnicas. Para entender la implementación del código será necesario explicar los comandos, variables y funciones utilizadas para la simulación.

El tipo de alcance que se aplica es descriptivo, el fenómeno sometido al análisis es el efecto del STO especificando sus propiedades y características. El ciclo continúa con el alcance correlacional donde hemos identificado las variables como BER y E_b/N_0 , su grado de asociación permite cuantificar el rendimiento de las diferentes técnicas de estimación para el efecto del STO y determinar la que tiene mejor desempeño entre las que se va a utilizar.

Finalmente, se tiene el alcance explicativo que permite identificar las causas del fenómeno y como se manifiesta al cambiar los parámetros de ciertas variables como: el tipo de modulación, factor de compresión y técnica de estimación.

2.1 PROGRAMA PRINCIPAL

En el programa principal SEFDM_varios_alfa_awgn se configuran los parámetros de entrada para la simulación. Por otra parte, en el Segmento de Código 2.1 se observan parámetros que se utilizan como datos para el transmisor que son: número de subportadoras, número de bits agrupados, números de ceros añadidos y factor de compresión. De esta manera, para obtener las gráficas se tiene las siguientes variables: el

número de iteraciones, Eb/No y número de símbolos por paquete. En la Tabla 2.1 se detallan las variables con su respectiva definición.

Tabla 2.1. Variables en el programa principal

Variable	Definición
N	Número de subportadoras
m	Número de bits agrupados según la modulación: 2: QPSK, 4:16-QAM, 6: 64-QAM
a	Número de ceros añadidos
alfa	Factor de compresión
Ntrials	Iteraciones para cada valor de SNR
EbNo	Valor de Eb/No
N_Sym	Numero de símbolos OFDM por paquete
BER	Tasa de Bits errados

Todas las variables en el Segmento de código 2.1 son definidas antes de la simulación Monte-Carlo, en este caso es el método para estimar el BER.

```

% INGRESO DE VARIABLES
%Datos del transmisor
N=16;           %Número de Subportadoras
m=4;           %Número de bits agrupados según la modulación
               % 2-->QPSK | 4--->16QAM | 6-->64QAM
a=[0,2]; %Número de ceros añadidos (Definición SEFDM)
alfa=(N./(N+a)); %Definición del Factor de Compresión de AB

%Datos para la obtención de gráficas
Ntrials=100;   %Número de iteraciones para cada valor de SNR
EbNo=0:2:30;   %Eb/No
N_sym=5;       %numero de simbolos OFDM por paquete

```

Segmento de código 2.1. Variables de ingreso para SEFDM_varios_alfa_awgn.m

La función SEFDM_awgn permite realizar la implementación del sistema SEFDM. En el Segmento de código 2.2 se observa las variables de entrada necesarias para el funcionamiento del sistema. En la Tabla 2.2 se colocan las variables y su tipo. Finalmente, el resultado de la función mencionada anteriormente es la tasa de bits errados.

```
function [BER] = SEFDM_awgn(N,m,alfa,Ntrials,EbNo,N_sym,nSTO,TecSTO,sincro)
```

Segmento de código 2.2. Definición de la función SEFDM_awgn.m

En este programa principal se utiliza la función SEFDM_awgn para implementar el transmisor, canal y receptor con su respectiva sincronización del Sistema SEFDM. El esquema del código implementado en dicha función se observa en la Figura 2.1. A continuación, se explicará en detalle cada uno de los componentes en las siguientes secciones.

Tabla 2.2. Variables de entrada y salida de la función SEFDM_awgn

Variables	Tipo de Variable
N	Entrada
m	Entrada
alfa	Entrada
Ntrials	Entrada
EbNo	Entrada
N_Sym	Entrada
BER	Salida

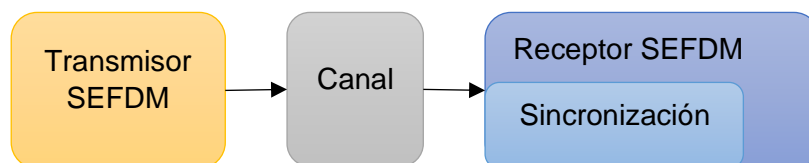


Figura 2.1. Esquema del código implementado

2.2 SISTEMA DE TRANSMISIÓN SEFDM

El sistema de transmisión como se puede observar en la Figura 2.2 se compone por varios bloques. El primero es la generación de bits aleatorios, los datos que van a ser transportados, seguido de eso se realiza la codificación convolucional para este caso con una tasa de codificación de $\frac{1}{2}$. A continuación, se tiene el bloque de modulación, y para el siguiente bloque se toman los valores con datos omitiendo los ceros agregados por el sistema SEFDM. Luego, se aplica la IFFT y se añade el prefijo cíclico. Finalmente, se añade

el preámbulo ya definido en una función específica. Todos estos bloques se encuentran implementados en la función SEFDM_awgn. Para esta sección se va a definir los bloques antes mencionados con especial énfasis en la creación e inserción del preámbulo.

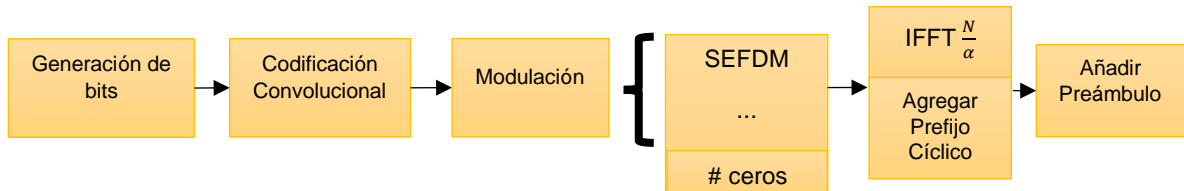


Figura 2.2. Transmisor SEFDM

2.2.1 VALORES INICIALES

En la tabla 2.3 se definen los valores iniciales tales como el orden de la modulación, tamaño de IFFT/FFT, SNR (*Signal to Noise Ratio*), la matriz para almacenar errores de cada repetición. Para la codificación es importante definir la tasa de codificación, parámetros del codificador FEC y objetos para el codificador convolutiva.

Tabla 2.3. Valores iniciales

Datos	Definición
M	Orden de la modulación
IFFT_FFT_size	Tamaño de la IFFT/FFT
Snr	Obtención del vector SNR
BER_T	Matriz para almacenar el BER
tc	Tasa de codificación
traceBack	Parámetro del codificador FEC
trellis	Parámetro por defecto
convEncoder	Características del codificador convolutiva
vitDecoder	Características del decodificador convolutiva

En el Segmento de Código 2.3 se observa el establecimiento de los valores iniciales mencionados y definidos anteriormente. Además, se definen los parámetros para el codificador convolutiva. Cabe recalcar que el objeto definido al final se utiliza en el sistema de recepción.

```

%VALORES INICIALES
M=2^m; %Orden de la modulación
IFFT_FFT_size=round(N/alfa); %Tamaño de la IFFT/FFT
Snr=EbNo+10*log10(m); %Obtención del vector SNR
BER_T=zeros(Ntrials,length(Snr)); %Creación de la matriz

%codificación
tc=1/2; %tasa codificación
traceBack = 32; %parametro de codificador FEC
trellis = poly2trellis(7,[171 133]);
%codificador convolucional
convEncoder = comm.ConvolutionalEncoder('TrellisStructure',trellis);
vitDecoder = comm.ViterbiDecoder('TrellisStructure',trellis, ...
    'InputFormat','unquantized','TracebackDepth',traceBack);

```

Segmento de código 2.3. Valores iniciales de la función SEFDM_awgn.m

En el Segmento del código 2.4 se observa como se obtiene la matriz de subportadoras. Para más información revisar en [10].

```

%MATRIZ DE SUBPORTADORAS F
F=zeros(N,N);
for k=0:N-1
    for n=0:N-1
        F(k+1,n+1)=1/sqrt(N)*exp(1i*2*pi*n*alfa*(k/N));
    end
end

%Matriz del Canal
H=F'*F;

```

Segmento de código 2.4. Definición de la matriz de subportadoras.

2.2.2 GENERACIÓN DE BITS

Al momento de generar los bits, hay que tomar en cuenta que son de carácter aleatorio en el rango de valores discretos entre $[0,1]$, el tamaño de los datos es $N*m*tc*N_{sym}$, lo cual se almacena en la variable `bit_tx_s`.

```

bit_tx_s=randi([0,1],N*m*tc*N_sym,1); %Generación de bits

```

Segmento de código 2.5. Generación de bits

2.2.3 CODIFICACIÓN

Se aplica la codificación convolucional con tasa $\frac{1}{2}$ a los bits generados aleatoriamente y se los almacena en la variable `data_cod`.

```
data_cod = convEncoder(bit_tx_s); %codificacion FEC
```

Segmento de código 2.6. Codificación

2.2.4 MODULACIÓN

Se pueden utilizar los diferentes esquemas de modulación como son QPSK, 16QAM y 64 QAM con un valor de m de 2, 4 y 6 respectivamente, lo cual representa el número de bits agrupado de la forma $M=2^m$.

```
%Modulación digital  
tx_mod=qammod(data_cod,M,'InputType','bit','UnitAveragePower',true);  
ceros=zeros(1,IFFT_FFT_size-N);  
tx_mod_s=tx_mod.';
```

Segmento de código 2.7. Modulación

2.2.5 IFFT

En la Ecuación 2.1 se obtiene el tamaño para la aplicación de la `ifft` y en la Ecuación 2.2 se calcula la cantidad de ceros que se agregarán a la señal modulada.

$$IFFT_FFT_size = \frac{N}{\alpha} \quad (2.1)$$

$$ceros = IFFT_FFT_size - N \quad (2.2)$$

La variable `tx_sefdm` almacena la señal modulada con los ceros incluidos como se puede observar en el Segmento de Código 2.8.

```
temp=tx_mod_s((jj-1)*N+1:jj*N);  
tx_sefdm=[temp zeros];
```

Segmento de código 2.8. Adición de ceros a la señal modulada

Para la conversión de la señal del dominio de la frecuencia al dominio del tiempo, se utiliza el comando `ifft` aplicado a la variable `tx_sefdm`, con el tamaño obtenido de la Ecuación 2.1. Por otro lado, en el Segmento de Código 2.9 se observa además que la variable `tx_out` contiene la señal de 1 hasta N es decir desconociendo los ceros agregados.

```
ifft_tx=ifft(tx_sefdm,IFFT_FFT_size); %Bloque IFFT  
tx_out=ifft tx(1:N); %Desconocer los últimos valores
```

Segmento de código 2.9. Aplicación de la IFFT

2.2.6 INSERCIÓN DEL PREFIJO CÍCLICO

Al insertar el prefijo cíclico de tamaño $\frac{N}{4}$ al inicio del símbolo SEFDM se evita el fenómeno de ISI. En el segmento de Código 2.10 se observa la inserción del prefijo cíclico.

```
tx_out_cp=[tx_out(N-(N/4)+1:end) tx_out];
```

Segmento de código 2.10. Inclusión del prefijo cíclico

2.2.7 PREÁMBULO

Para la implementación de dos técnicas de estimación del STO se requiere añadir el preámbulo a la señal. El preámbulo se caracteriza por ser una secuencia conocida tanto en el transmisor como en el receptor y se lo obtiene por medio de la función `getpream`. En dicha función el único parámetro de entrada es el número de subportadoras.

En el Segmento de Código 2.11 se observa como se utiliza la función y luego se reorganiza el preámbulo obtenido. El tamaño del preámbulo se almacena en la variable `length_pream`. En la variable `tx_out_s` se guarda la señal con el preámbulo insertado al principio.

```
%Preámbulo
[pream]=getpream(N);           %Creación del preámbulo de la forma PC PC A A-A-A
pream_TX = pream';
length_pream=length(pream_TX); %Tamaño del preámbulo
tx_out_s=[pream_TX tx_out_s];
```

Segmento de código 2.11. Adición del preámbulo a la señal

En el Segmento de Código 2.12 se observa la implementación del preámbulo iniciando con la secuencia de ruido PN (*Pseudo Noise Sequence*), luego el envío de ceros en las frecuencias pares. A continuación, se cambia al dominio del tiempo y finalmente se arma el preámbulo final que se va a explicar más adelante.

```
function [pream] = getpream(IFFT_FFT_size)
%Secuencia de Pseudo Ruido
PN_seq = [1 1 -1 1 -1 -1 1 -1 1 1 -1 1 -1 -1 -1];
PN_seq(2:2:end)=0;           %Envío de ceros en frecuencias pares
PN_seq_ord=zeros(IFFT_FFT_size,1); %Inicializar variable
PN_seq_ord(1:16)=PN_seq(1:16);
pre1=ifft(PN_seq_ord, IFFT_FFT_size); %Dominio del tiempo
pc=0*pre1(1:4);
[pream]=[pc;pc;pre1;-pre1];%Preámbulo de la forma [PC PC A A -A -A]
end
```

Segmento de código 2.12. Función que permite obtener el preámbulo

Para empezar con la estructura del preámbulo es necesario definir como se componen los símbolos, dicho símbolo está conformado por una estructura repetitiva. Luego de cambiar al dominio del tiempo se obtienen los bloques A para completar la estructura del Símbolo 1. A continuación, se detalla como se define inicialmente el preámbulo.

En primer lugar, como se observa en la Figura 2.3 se envían ceros en las frecuencias pares en el símbolo que contiene la secuencia de Pseudo Ruido. Luego se ordena la secuencia y se almacena en la variable `PN_seq_ord`.

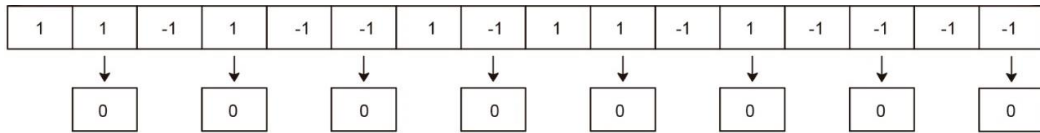


Figura 2.3. Agregar ceros al Símbolo 1.

En la Figura 2.4 se aplica la IFFT al preámbulo con los ceros agregados. Se realiza también el proceso de la IFFT para pasar el preámbulo al dominio del tiempo, en otras palabras, se tienen números complejos en los datos. Luego de pasar por el proceso se obtiene el símbolo 1 compuesto por dos bloques A.

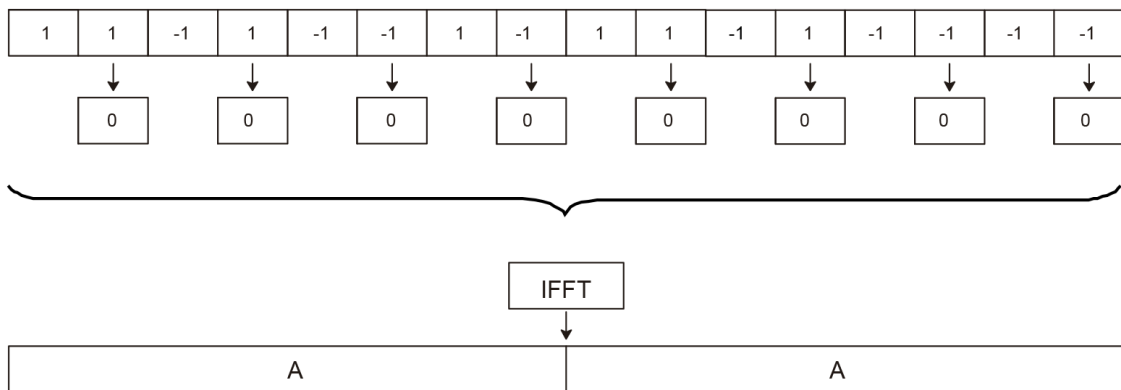


Figura 2.4. Aplicación de la IFFT.

Finalmente, se agrega un tiempo de guarda igual al tamaño de dos prefijos cíclicos para evitar que se obtengan valores erróneos de STO. Además, se utiliza otro símbolo que contiene la misma información del primer símbolo en versión negativa con la finalidad de mejorar la estimación del inicio del símbolo. En la Figura 2.5 se tiene la estructura final del preámbulo y se enfatiza en esta parte, ya que dos técnicas de estimación utilizan dicho preámbulo para estimar el inicio del símbolo.

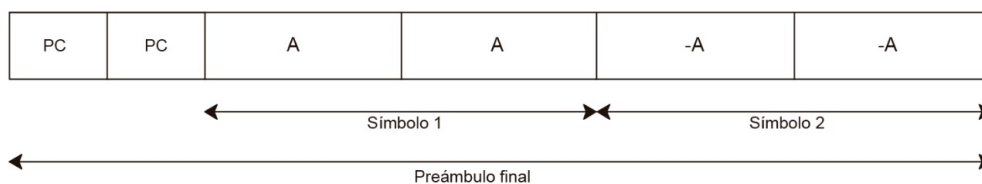


Figura 2.5. Estructura final del preámbulo

En la Figura 2.6 se observa el valor absoluto del preámbulo graficado, pudiéndose apreciar el tiempo de guarda formado por el tamaño de dos prefijos cíclicos, y a continuación los datos repetitivos. Hay que recalcar que se muestran valores absolutos del preámbulo y que se encuentra en el dominio del tiempo.

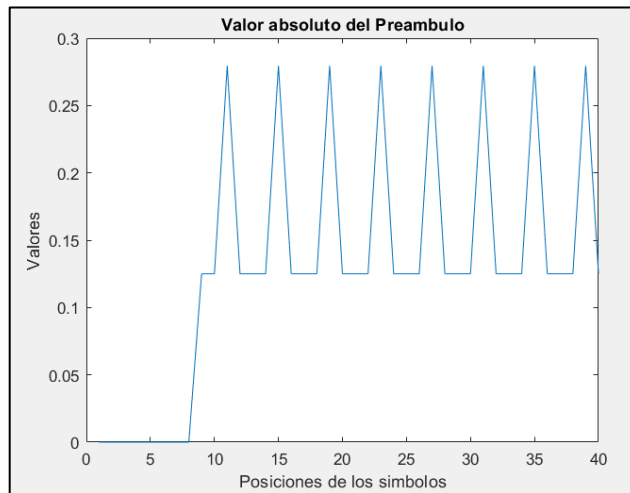


Figura 2.6. Valor absoluto del preámbulo graficado

2.3 CANAL AWGN

Para el canal inalámbrico se adiciona ruido blanco Gaussiano, por medio del comando `awgn`, que requiere como parámetros de entrada la señal transmitida y la relación señal a ruido. En el Segmento de Código 2.13 se observa que los datos que serán sumados con ruido se almacenan en la variable `canal_rx`.

```
canal_rx=awgn(tx_out_s,Snr(i),'measured','db');
pt=mean(abs(tx_mod_s.^2)); % se obtiene la media
```

Segmento de código 2.13. Ruido Blanco Gaussiano añadido a la señal

2.4 SISTEMA DE RECEPCIÓN SEFDM

Para el receptor se realiza primero la sincronización de la señal, extracción del preámbulo, remoción del prefijo cíclico y FFT. A continuación, la demodulación y decodificación como se observa en la Figura 2.7.

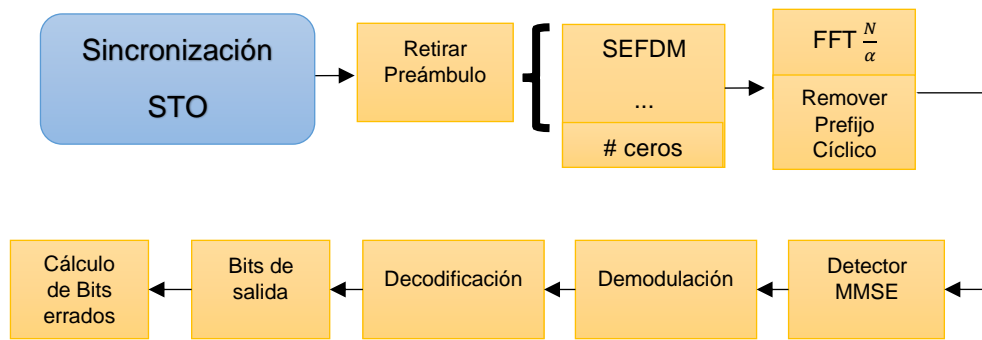


Figura 2.7. Receptor del sistema SEFDM

Para realizar el proceso de sincronización es importante identificar las variables y desplazamiento de STO que se agrega a la señal, la variable $TecSTO$ tiene el objetivo de permitir al usuario seleccionar la técnica de sincronización que se vaya a utilizar. En el Segmento de Código 2.14 se observan los datos necesarios para poder realizar la sincronización y aplicar el efecto del STO.

```

% Sincronización STO
TecSTO=3;      % Tecnica de estimación
               % 1--> Prefijo ciclico
               % 2--> Preambulo diferencia misma senial con ruido
               % 3--> Preambulo diferencia usando el preambulo sin ruido
nSTO=4;       % Desplazamiento en tiempo STO
Sinc_on=1;    % se utiliza para obtener el BER con sincronización
Sinc_off=0;   % se utiliza para obtener el BER sin sincronización

```

Segmento de código 2.14. Variables utilizadas para sincronización

La señal de entrada después de pasar por el canal AWGN se almacena en la variable `canal_rx_STO`. Para obtener un resultado con sincronización perfecta se utiliza la variable `Sincro` se puede tener un valor de: 0 sincronización perfecta y 1 usando técnica de sincronización. Luego se estima la posición del símbolo y al final se almacena la señal sincronizada en la variable `canal_rx_STO_comp`. En la Tabla 2.4 se observan las variables necesarias para realizar el proceso de sincronización.

Tabla 2.4. Variables para la sincronización

Variable	Descripción
canal_rx_STO	Señal a través del canal
N_Sym	Número de símbolos
TecSTO	Técnica de estimación
nSTO	Desplazamiento en tiempo STO
Sincro	Sincronización activada o desactivada. Sinc_on=1,Sinc_off=0
length_pream	Longitud del preámbulo
canal_rx_STO_comp	Señal sincronizada

2.4.1 SINCRONIZACIÓN SEFDM

Como se expone en la Figura 2.8, para el proceso de sincronización es necesario realizar tres pasos importantes: la inserción que permite añadir el efecto de STO, estimación donde se aplican las técnicas para determinar el inicio del símbolo y como paso final la compensación de STO.

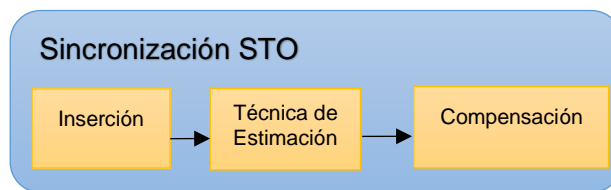


Figura 2.8. Sincronización del sistema SEFDM

2.4.1.1 Inserción de STO

Al ser recibida la señal, es necesario insertar el efecto de STO. Esto es posible por medio del desplazamiento STO en tiempo, y se aplica en base a la variable nSTO que puede tomar valores positivos y negativos. Dependiendo de su valor se agregarán ceros al final o al inicio del símbolo, y se almacena en la variable canal_rx_STO como se puede observar en el Segmento de Código 2.15.

```

%Inserción del STO
if nSTO>=0
    canal_rx_STO=[canal_rx(nSTO+1:end) zeros(1,nSTO)];% CASO 4
else
    canal_rx_STO=[zeros(1,-nSTO) canal_rx(1:end+nSTO)]; % CASO I y III
end

```

Segmento de código 2.15. Inserción del STO

En la Figura 2.9 a) se observa que si la variable nSTO toma un valor negativo se agregarán los ceros al inicio de la señal transmitida, pero si el valor colocado en la variable es cero quiere decir que coincide con el inicio del símbolo. En cambio, en la Figura 2.9 b) la condición es que si el desplazamiento es positivo lo ceros se agregan al final de la señal. Lo anteriormente comentado explica que en el fenómeno del STO es posible tener varios casos donde el símbolo llega antes, después o coincide con el inicio del símbolo.

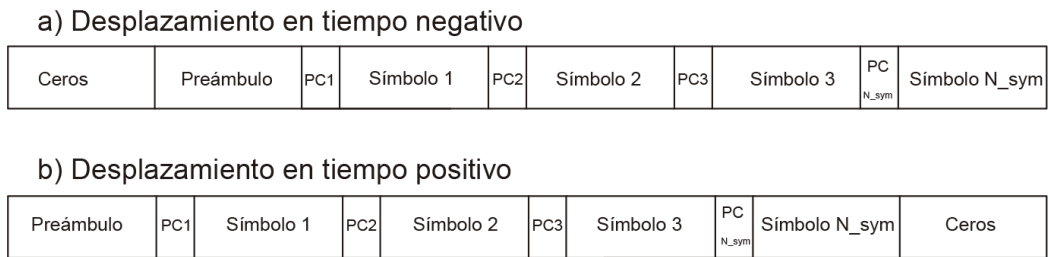


Figura 2.9. Inserción del desplazamiento en tiempo

2.4.1.2 Estimación de STO

Las técnicas empleadas en este trabajo se detallan en la Tabla 2.5. Las técnicas revisadas en el Capítulo 1 reciben la señal almacenada en la variable canal_rx_STO, recordando que esta señal ya posee el efecto del STO.

Tabla 2.2. Técnicas de estimación STO

N°	Técnica de Estimación
1	Estimación STO utilizando la diferencia mínima del Prefijo Cíclico
2	Estimación STO utilizando la diferencia mínima del Preámbulo
3	Estimación STO utilizando la diferencia mínima del Preámbulo comparado con el preámbulo conocido en el transmisor

Para las técnicas implementadas se utiliza una estructura de control que permita escoger que técnica va a estimar el inicio del símbolo, como se observa en el Segmento de Código 2.16.

```

%Técnicas de Estimación
if TecSTO == 1
    STO_est= STO_diferencia_PC(canal_rx_STO,N,N_sym);

elseif TecSTO == 2
    STO_est= STO_diferencia_pream(canal_rx_STO,length_pream);

elseif TecSTO == 3
    STO_est=STO_diferencia_pream_v2(canal_rx_STO,pream_TX,length_pream);
end

```

Segmento de código 2.16. Condicional para seleccionar la técnica

Para el proceso de estimación STO se utilizaron tres técnicas que son por medio del: prefijo cíclico, preámbulo y preámbulo conocido en el transmisor. Estas técnicas utilizan la señal en el dominio del tiempo para la estimación y se explican a continuación [8].

ESTIMACIÓN STO UTILIZANDO LA DIFERENCIA MÍNIMA DEL PREFIJO CÍCLICO

En la Tabla 2.3 se definen y se especifica el tipo de dato de entrada y salida que se obtiene de la función implementada para la estimación utilizando el prefijo cíclico. Se requiere la señal con el efecto del STO, el número de símbolos y las subportadoras para finalmente dar como resultado la estimación del inicio del símbolo.

Tabla 2.3. Variables para la técnica de estimación Prefijo Cíclico

Datos	Tipo de Dato	Descripción
canal_rx_STO	Entrada	Señal con efecto STO
N_Sym	Entrada	Número de símbolos
N	Entrada	Número de subportadoras
STO_est	Salida	Valor estimado

La primera técnica implementada utiliza el prefijo cíclico, por medio de la Ecuación (1.1) definida en el Capítulo 1. La estimación del STO se realiza mediante la diferencia de dos ventanas. La función se define como STO_diferencia_PC, que en primera instancia define un tamaño de ventana igual al prefijo cíclico. Luego la variable Nofdm que almacena la suma del tamaño de ventana de la FFT o IFFT con el tamaño del prefijo cíclico.

La variable com_delay se coloca un valor de entre 1 y 40 que corresponde al tamaño del preámbulo con el fin de no recorrer todas esas posiciones se establece en 34 el inicio de

la comparación. Se compensa el desfase de inicio en la variable `STO_est` adicionando siete a la posición estimada. Se realizan también comparaciones hasta el número de símbolos menos uno, almacenados en la variable `Num_CP` con la finalidad de no sobrepasar el tamaño de los datos de la señal. El proceso que se realiza se observa en la Figura 2.10 y la implementación en el Segmento de Código 2.17.

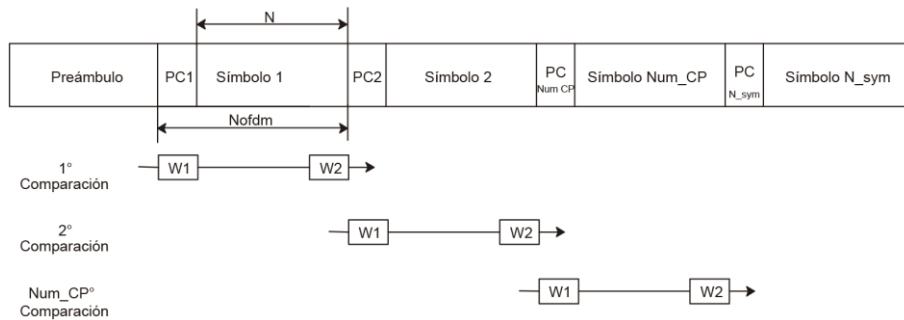


Figura 2.10. Estimación usando el Prefijo Cíclico

En la Figura 2.11 se observa como se realiza la primera comparación almacenando en la variable `tmp` la diferencia de la misma señal, los valores colocados en cada recuadro representan las posiciones que se toman de la señal `canal_rx_STO`. Las ventanas son de tamaño W , en la primera iteración la ventana tiene las posiciones de la 35 hasta la 38 y se resta con la ventana que tiene las posiciones de la 51 hasta la 54. La siguiente iteración se desplaza en una posición. Entonces, ahora la primera ventana se ubica en la posición 36 hasta la 39 y se resta con la ventana que va de la posición 52 hasta la 55, y así hasta llegar a la última iteración.

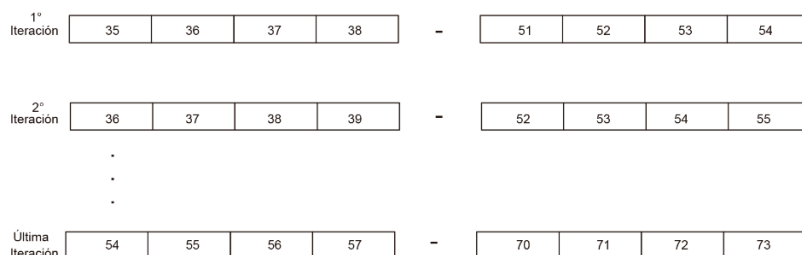


Figura 2.11. Iteraciones para encontrar la diferencia mínima

```

function STO_est= STO_diferencia_PC(canal_rx_STO,FFT_IFFT_size,N_sym)
%Técnica de estimación mínima diferencia utilizando Prefijo Cíclico
W=FFT_IFFT_size/4; %Tamaño de ventana igual al PC (4)
Nofdm=FFT_IFFT_size+W; %Longitud que corre la ventana (20)
com_delay=34; %Punto de partida de la ventana para evitar el preámbulo
Num_CP=N_sym-1; %Número de comparaciones (4)

%Inicializacion
Sum=zeros(Num_CP,Nofdm);
Sum_t=zeros(1,Nofdm);
m = 0;
STO_est=0;

while m<Num_CP-1 %Repite para comparar el prefijo cíclico de cada símbolo
%Desplazamiento de ventana
for i=1:Nofdm
nn=i+com_delay+(0:W-1);
tmp=canal_rx_STO(nn+m*Nofdm)-canal_rx_STO(nn+FFT_IFFT_size+m*Nofdm);
Sum(m+1,i) = sum(abs(tmp)); %ecuación Diferencia mínima PC 1.1

end
Sum_t=Sum(m+1,:)+Sum_t; %Máxima similitud
m=m+1;
end
[Mag, Pos]=min(Sum_t); %Valor y posición del mínimo valor
STO_est = -Pos+7; %Estimación STO
end

```

Segmento de código 2.17. Implementación usando el Prefijo Cíclico

La variable Sum permite almacenar los valores encontrados anteriormente. Al culminar el ciclo iterativo se almacena la suma de cada diferencia en la variable Sum_t. Al finalizar el ciclo de decisión se obtiene la magnitud y posición por medio de la función min de la variable Sum_t, recordando que se tiene que compensar la posición adicionando un valor específico se obtiene en la estimación STO dentro de la variable STO_est. Además, el valor específico se lo obtiene restando la longitud total del preámbulo con la variable com_delay y adicionando uno al resultado.

ESTIMACIÓN STO UTILIZANDO LA DIFERENCIA MÍNIMA DEL PREÁMBULO

Para esta técnica en la Tabla 2.4 se detallan las variables de ingreso y salida. Esta técnica a diferencia con la anterior va a comparar el preámbulo, y para ello es necesario su longitud y la señal con el efecto del STO. Además, se va a comparar el preámbulo de la misma señal por medio de la diferencia mínima.

Tabla 2.4. Variables para la técnica de estimación Preámbulo

Datos	Tipo de Dato	Descripción
canal_rx_STO	Entrada	Señal con efecto STO
length_pream	Entrada	Longitud del preámbulo
STO_est	Salida	Valor estimado

La siguiente técnica que se utiliza para la estimación es por medio del preámbulo y utiliza la Ecuación (1.4) del Capítulo 1. El desplazamiento en tiempo se estima por medio de la diferencia mínima entre dos símbolos iguales dentro del preámbulo. Se crea la función STO_diferencia_pream y se declara primero el tamaño de la ventana, el cual divide el tamaño total del preámbulo para 5 símbolos.

Es decir que el preámbulo contiene cuatro símbolos repetitivos, dos positivos y los otros dos negativos. El punto de partida se define en 0, pues inicia desde el comienzo de todos los datos, sin embargo, hay que compensar los ceros que contiene el preámbulo o tiempo de guarda como se explicó en la sección del preámbulo. A la variable Pos se le adiciona nueve, ya que así se obtiene la estimación correcta del STO. El proceso se puede observar en la Figura 2.12 y la implementación en el Segmento de Código 2.18.

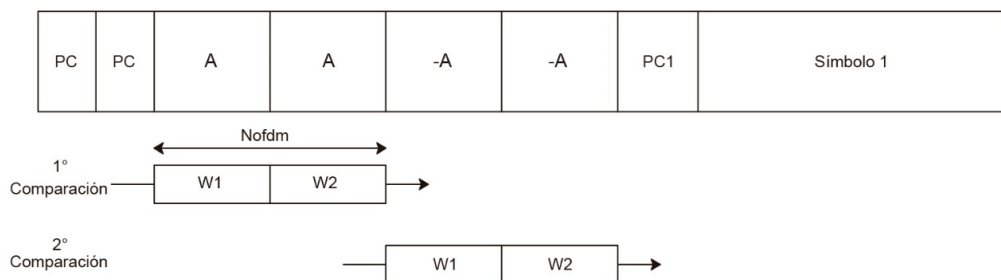


Figura 2.12. Estimación usando el Preámbulo

```

function STO_est=STO_diferencia_pream(rxSig_STO,length_pream)
%Técnica de estimación mínima diferencia utilizando Preámbulo
W=length_pream/5;      %tamaño de la ventana 8
Nofdm=W+W;            %Logitud que recorre 16
com_delay=0;          %Punto de partida de la ventana

%Inicializacion
Sum =zeros(2,Nofdm);
Sum_t= zeros(1,Nofdm);
STO_est=0;
m=0;
for m=0:1             %Comparación de los símbolos repetitivos del preámbulo A&-A
    %Deslizamiento de ventanas
    for i=1:Nofdm
        nn=i+com_delay+(0:W-1);
        tmp =rxSig_STO(nn+2*W*m)-rxSig_STO(nn+W+2*W*m);
        Sum(m+1,i)=sum(abs(tmp)); %ecuacion Diferencia mínima preámbulo
    end
    Sum_t = Sum(m+1,:)+Sum_t;      %Almacena valor de cada diferencia
end
[Mag, Pos]=min(Sum_t);           %Valor y posición del mínimo valor
STO_est = -Pos+9;                %Estimación STO
end

```

Segmento de código 2.18. Implementación usando el Preámbulo

En la Figura 2.13 se expone la primera y segunda comparación para el Símbolo 1 que se encuentra dentro del preámbulo de la misma señal. Los números que se observan en cada recuadro corresponden a las posiciones de la señal. En la comparación del Símbolo 1 las ventanas son de tamaño W , en la primera iteración la ventana va de la posición 1 hasta la 8 y se resta con la ventana que toma las posiciones desde la 9 hasta la 16. En la segunda iteración, la ventana se desplaza una posición es decir la ventana inicia en la posición 2 hasta la 9 y se resta con la ventana desde la posición 10 hasta la 17, este movimiento de ventana se realiza hasta llegar a la última iteración.

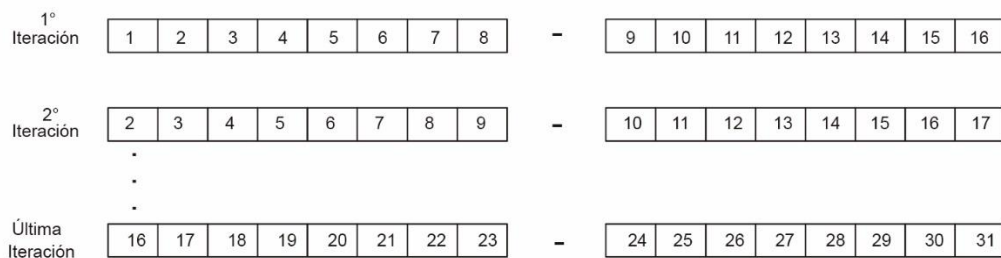


Figura 2.13. Comparación Símbolo 1

En la Figura 2.14 se observa la comparación de Símbolo 2, la primera iteración la ventana se define desde la posición 17 hasta la 24 y se resta con la ventana definida desde la posición 25 hasta la 32. La segunda iteración de igual forma se desplaza en una posición, la ventana va desde la posición 18 hasta la 25 y se resta con la ventana que va desde la posición 26 hasta la 33, este proceso se realiza hasta llegar a la última iteración.

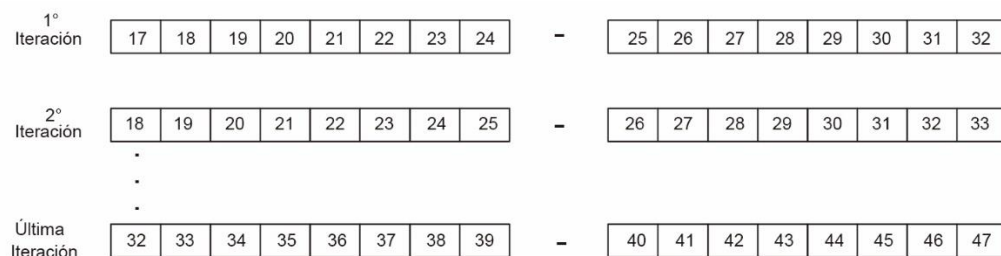


Figura 2.14. Comparación Símbolo 2

Para el Símbolo 1 la diferencia obtenida se almacena en la variable Sum, del mismo modo se realiza para el Símbolo 2 del preámbulo. Finalmente, la variable Sum_t almacena la

suma de todas las diferencias obtenidas en las comparaciones. De igual forma que en la técnica anterior se obtiene la Posición y se la compensa adicionando 9.

El valor anterior se lo obtiene de la suma de las longitudes de los dos prefijos cíclicos más uno, recordando que el preámbulo está compuesto inicialmente por este tiempo de guarda.

ESTIMACIÓN STO UTILIZANDO LA DIFERENCIA MÍNIMA DEL PREÁMBULO COMPARADO CON EL PREÁMBULO CONOCIDO EN EL TRANSMISOR

La siguiente técnica de Estimación se diferencia de la anterior en que además de ingresar la señal con el efecto STO, ingresa el preámbulo conocido tanto en el trasmisor, como en el receptor. Para la diferencia mínima se utiliza la señal con el efecto del STO y el preámbulo se mantiene en la posición constante para el primer símbolo y luego para el segundo símbolo, ambos contenidos dentro del preámbulo. En la Tabla 2.5 se detallan los datos utilizados para su funcionamiento.

Tabla 2.5. Variables para la Técnica de Estimación Preámbulo conocido en el transmisor

Datos	Tipo de Dato	Descripción
canal_rx_STO	Entrada	Señal con efecto STO
length_pream	Entrada	Longitud del preámbulo
pream_TX	Entrada	Preámbulo conocido en el transmisor
STO_est	Salida	Valor estimado

Finalmente, se tiene esta técnica de estimación que utiliza la Ecuación (1.5) del Capítulo 1. Para este caso se estima el desplazamiento en tiempo por medio de la diferencia mínima entre la señal recibida con el efecto STO y el preámbulo conocido en el transmisor y receptor.

Se utiliza la función STO_diferencia_pream_v2 que inicialmente declara un tamaño de ventana igual al tamaño de un Símbolo (A, A) que contiene el preámbulo. Se mantiene el preámbulo estático en cada una de las comparaciones como se observa en la Figura 2.15.

Al igual que la técnica anterior, las comparaciones de la ventana inician desde 0 definido en la variable com_delay y se compensan de igual forma que la técnica anterior. La implementación se la puede observar en el Segmento de Código 2.19.

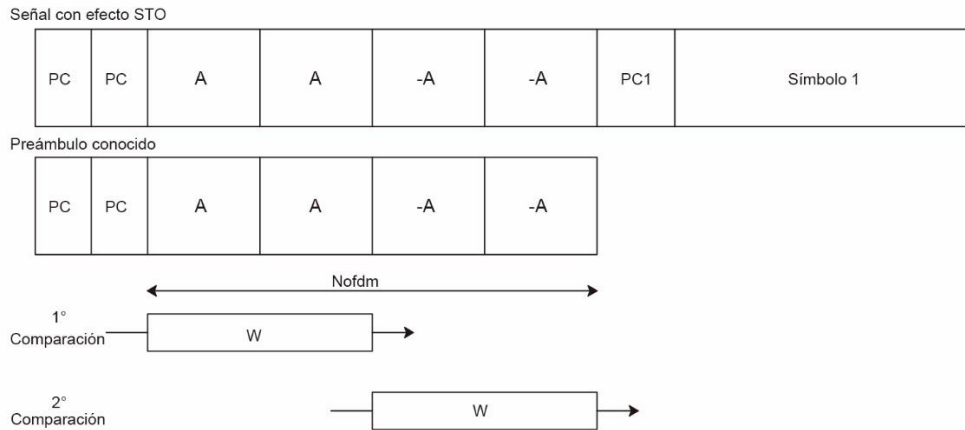


Figura 2.15. Estimación usando el Preámbulo conocido en el transmisor

En la Figura 2.16 se tiene las posiciones que se toma de la señal recibida y del preámbulo conocido. Al igual que la técnica anterior se observa la comparación, pero la diferencia es que se mantiene fija la posición del preámbulo conocido en el transmisor y receptor. En la primera iteración la ventana va desde la posición 1 hasta la 16 y se resta con la ventana fija del preámbulo conocido desde la posición 9 hasta la 24. En la segunda iteración la ventana se desplaza en una posición y se establece desde la posición 2 hasta la 17 y se resta con la ventana fijada del preámbulo conocido. Este proceso se realiza hasta llegar a la última iteración.

```
function STO_est=STO_diferencia_pream_v2(rxSig_STO,pream_TX,length_pream)
    %Técnica de estimación mínima diferencia utilizando Preámbulo
    W = 16;           %tamaño de la ventana 16
    Nofdm=W+W;       %Logitud que recorre 32
    com_delay=0;     %Punto de partida de la ventana

    %Inicializacion
    Sum =zeros(2,Nofdm);
    Sum_t= zeros(1,Nofdm);
    STO_est=0;
    m=0;

    for m=0:1        %Comparación de los símbolos repetitivos del preámbulo A&-A
        %Desplazamiento de ventanas
        for i=1:Nofdm
            nn=i+com_delay+(0:W-1);
            tmp =rxSig_STO(nn+2*W*m)-pream_TX(9+m*16:9+W-1+m*16);
            Sum(m+1,i)=sum(abs(tmp)); %Ecuación Diferencia mínima
        end
        Sum_t = Sum(m+1,:)+Sum_t; %Almacena valor de cada diferencia sumada
    end
    [Mag,Pos]=min(Sum_t); %Valor y posición del mínimo valor
    STO_est = -Pos+9; %Estimación STO
end
```

Segmento de código 2.19. Implementación usando el preámbulo conocido

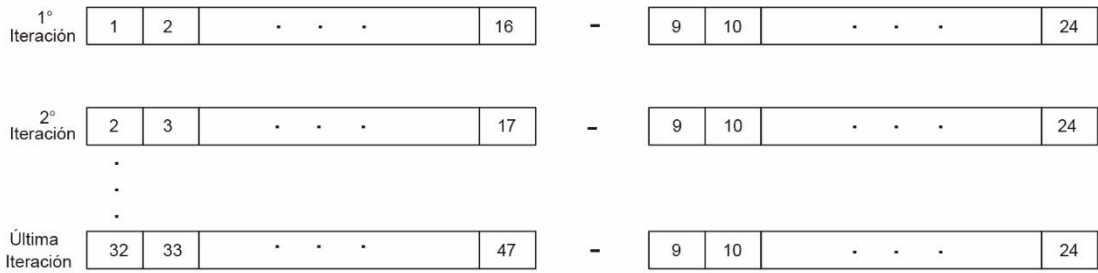


Figura 2.16. Comparaciones con el Preámbulo conocido en el transmisor

Para el Símbolo 2 se realiza el proceso similar al de la anterior técnica. La diferencia es el tamaño de la ventana y la ventana fija que se establece en el preámbulo conocido que ahora iría de la posición 25 hasta la 40. Al hacer la diferencia primero se la realiza para el Símbolo 1 del preámbulo, y una vez que finaliza la secuencia iterativa se realiza para el Símbolo 2. Se utiliza la variable Sum para almacenar las diferencias. La variable Sum_t se utiliza para guardar el total de diferencias y así determinar la posición al igual que la técnica anterior.

2.4.1.3 Compensación del STO

Luego de estimar el desplazamiento en tiempo se realiza la compensación que básicamente es el proceso contrario a la inserción del STO. Una vez que se obtiene la estimación en la variable STO_est, se añaden ceros al inicio o al final del símbolo es decir a la señal contenida en la variable canal_rx_STO como se observa en el Segmento de código 2.20.

```

%Compensación
if STO_est >= 0
    canal_rx_STO_comp=[zeros(1,STO_est) canal_rx_STO(1:end-STO_est)];
else
    canal_rx_STO_comp=[canal_rx_STO(-(STO_est-1):end) zeros(1,-STO_est)];
end

```

Segmento de código 2.20. Compensación del STO

2.4.2 RETIRAR PREÁMBULO

Una vez que se ha sincronizado la señal y se ha estimado el inicio del símbolo se procede a retirar el preámbulo añadido en el transmisor. En el Segmento de Código 2.21 se retira el preámbulo tomando la longitud de este sumado uno, para que sea la siguiente posición es decir que solo contenga los cinco símbolos y sus correspondientes prefijos cíclicos.

```
% Retirar preámbulo con sincronización
canal_rx=canal_rx_STO_comp((length_pream+1):end);
```

Segmento de código 2.21. Extracción del preámbulo

Para el caso de la señal que se omite la sincronización se evita todo el proceso del STO y se procede a solamente retirar el preámbulo con la ayuda de la estructura de control como se observa en el Segmento de Código 2.22.

```
else
    % Retirar preámbulo sincronización perfecta
    canal_rx=canal_rx((length_pream+1):end);
end
```

Segmento de código 2.22. Extracción del preámbulo para sincronización perfecta

2.4.3 RETIRAR PREFIJO CÍCLICO / FFT

En el segmento de código 2.23 se observa que primero se remueve el prefijo cíclico, es decir las $\frac{N}{4}$ muestras que se colocaron al inicio de cada símbolo y se almacena en la variable temp. Luego se añaden los ceros como definición SEFDM que se obtiene en la Ecuación (2.2), estos representan los ceros a añadirse a los datos almacenados en la variable temp. A continuación, se aplica la FFT con el tamaño de ventana correspondiente que posibilita su cambio al dominio de la frecuencia. Finalmente, se descartan los ceros agregados por el sistema SEFDM.

```

temp=canal_rx((jj-1)*(N+N/4)+1+N/4:jj*(N+N/4));
rx_sefdm_vec=[temp,ceros];           %Definición FFT para SEFDM
fft_rx=fft(rx_sefdm_vec,IFFT_FFT_size); %Bloque FFT
Rx=fft_rx(1:N);                     %Desconocer los últimos valores

```

Segmento de código 2.23. Remoción prefijo cíclico y bloque FFT

2.4.4 DETECCIÓN

Para este trabajo se utilizó el detector MMSE que tiene como objetivo cancelar el ICI, para este caso se utilizará la función MMSE. En el segmento de código 2.24 se observa la función utilizada para el detector donde los datos que se almacenan en la variable rx_det.

```

I=eye(N); %Matriz Identidad
rx_det=MMSE(Rx_p,I,Snr(i),H,pt/(N*4)); %Detector MMSE

```

Segmento de código 2.24. Detección MMSE

2.4.4.1 MMSE

En la tabla 2.6 se observan las variables y tipos de datos que recibe la función con la finalidad de obtener la señal detectada con MMSE.

Tabla 2.6. Variables utilizadas en la función MMSE

Datos	Tipo de Dato	Descripción
R	Entrada	Señal de entrada
I	Entrada	Matriz identidad
snrdb	Entrada	Valor de SNR en dB
H	Entrada	Matriz del canal
pt	Entrada	Potencia promedio de la señal recibida
Sv	Salida	Señal detectada con MMSE

Se calcula en la variable snr el valor de SNR en veces, en la variable W se almacenan los datos aplicando la definición de detector MMSE, y en la variable Sv se multiplica la definición del detector MMSE con la señal de datos para así obtener los símbolos. En el

segmento de código 2.25 se observa su implementación; para información a detalle revisar [10].

```
function [Sv] = MMSE(R,I,snrdb,H,pt)
    snr=10^(snrdb/10);           %Valor de SNR en veces
    W=((H'*H)+(pt/snr)*I)\H';    %Aplicación de la definición
                                %del detector MMSE
    Sv=W*R;                     %Obtención de símbolos
end
```

Segmento de código 2.25. Implementación del Detector MMSE

2.4.5 DEMODULACIÓN

El objetivo principal de este apartado es reconstruir la señal, modulada en el transmisor. Se utiliza la función `qamdemod`, donde se especifica la señal a ser demodulada, el orden de la modulación $M=2^m$, y el tipo de datos de salida. En el segmento de Código 2.26 se observa el desarrollo del proceso de demodulación.

```
%Demodulación digital
rx_dem=qamdemod(rx_det_s,M,'OutputType','approx11r','UnitAveragePower',true,'NoiseVariance',noiseVar);
```

Segmento de código 2.26. Proceso de Demodulación

2.4.6 DECODIFICACIÓN

En el segmento de código 2.27 se observa que para la decodificación FEC se utiliza el objeto creado en los parámetros iniciales que se almacena en la variable `vitDecoder`. Al ser un codificador convolucional de tasa $\frac{1}{2}$ se aplica redundancia con la finalidad de reducir el número de errores al calcular el BER.

```
rx_bit_dec = vitDecoder(rx_dem);           %decodificacion FEC
```

Segmento de código 2.27. Decodificador FEC

2.4.7 CÁLCULO DE BITS ERRADOS

Finalmente, lo ideal sería que el receptor obtenga datos idénticos a los enviados desde el transmisor, en contraste, se desea determinar el efecto STO que el sistema SEFDM sufre, como por ejemplo ciertas variaciones en la estimación de inicio del símbolo provocados por la naturaleza del canal inalámbrico. Se usa el comando biterr para obtener el número de errores.

El comando biterr utiliza la comparación de dos vectores de igual tamaño, el primer vector es la señal transmitida y se compara con la señal demodulada, almacenadas en las variables bit_tx_s y rx_bit_dec respectivamente. En la variable n_error se almacena el número de errores y en la variable ratio se tiene la tasa de errores que se almacena en la matriz BER_T.

Al terminar las iteraciones se extrae el valor promedio para cada SNR por medio de la función mean. La variable BER almacena dicho valor como dato de salida de la función SEFDM_awgn. En el Segmento de Código 2.28 se observa como se obtiene la tasa de bits errados.

```
        %Número de Errores
        [n_error, ratio]=biterr(bit_tx_s(1:end-traceBack), rx_bit_dec(traceBack+1:end));

        %Matriz de tasa de errores
        BER_T(j,i)=ratio;
    end
end
%Promedio de tasa de Errores
BER=mean(BER_T);
end
```

Segmento de código 2.28. Promedio de la tasa de bits errados

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

En este capítulo se presentan los resultados de las simulaciones en el sistema SEFDM con un canal AWGN y con técnicas de estimación STO implementadas en el receptor. Para su análisis se utiliza las gráficas de BER vs Eb/No. En dichas gráficas se decidió variar el factor de compresión, el tipo de modulación, el tipo de técnica y los valores de desplazamiento en tiempo positivo o negativo.

3.1 ESCENARIO DE SIMULACIÓN

El sistema SEFDM implementado se caracteriza por: codificación convolucional de tasa $\frac{1}{2}$, la señal atraviesa un canal AWGN, se utiliza un tamaño de 16 para la ventana de la FFT y detector MMSE para obtener la señal recuperada. En el programa principal se puede variar el tipo de modulación siendo posibles las siguientes: QPSK, 16QAM o 64QAM, para el factor de compresión de ancho de banda se utiliza los siguientes valores: alfa {0.67,0.84,0.89,0.94}, el número de iteraciones y el valor de Eb/No. En este caso se hace énfasis en la sincronización del STO para lo cual se tiene 3 técnicas de estimación y un rango de [-4,4] para el desplazamiento en tiempo.

Para el análisis correspondiente se necesita las gráficas de BER vs Eb/No. Para obtener las gráficas se utiliza en el programa principal 5000 a 10000 iteraciones con el fin de obtener gráficas con mayor exactitud.

3.2 PRUEBAS DEL SISTEMA

Para comprobar el correcto funcionamiento se realiza una simulación con el factor de compresión alfa igual a 1, con la finalidad de graficar un sistema sin compresión de ancho de banda es decir OFDM. La tendencia se comprueba ya que para un determinado valor Eb/No la curva que representa la modulación QPSK presenta menor valor de BER comparado con los otros tipos de modulaciones.

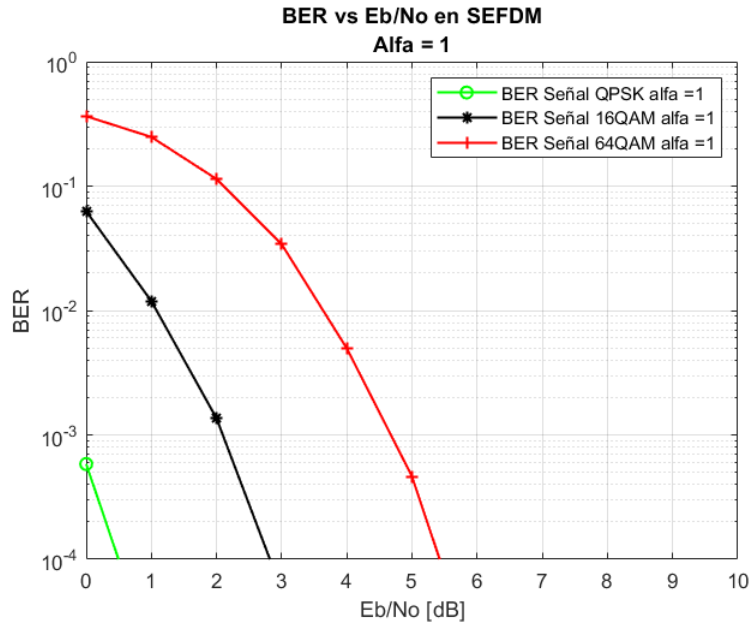


Figura 3.1. BER vs Eb/No de un Sistema SEFDM con alfa igual a 1

3.3 RESULTADOS

En este subcapítulo se tiene los resultados de la simulación de las tres técnicas de estimación STO recordando que se utilizó el prefijo cíclico, preámbulo y preámbulo conocido.

En la Figura 3.2 se utiliza la modulación QPSK con un factor de compresión de 0.67 y desplazamiento en tiempo positivo igual a 4. En la gráfica se observa las curvas de estimación que se comparan con la de sincronización perfecta que es la de color verde. Para la estimación del prefijo cíclico y preámbulo se aprecia que para 3 dB en Eb/No son similares. En cambio, la estimación con preámbulo conocido es la que mejor desempeño tiene y es muy similar a la sincronización perfecta.

En la Tabla 3.1 se detalla los valores obtenidos del gráfico, para un Eb/No en específico, el valor de BER obtenido es menor en la técnica de preámbulo conocido. Sin embargo, al incrementar el Eb/No a 7 dB la técnica del preámbulo se asemeja bastante con la del preámbulo conocido. En todo el intervalo se considera que la técnica del preámbulo conocido en el transmisor es la que tiene el mejor rendimiento. Algo que se puede notar en la técnica del preámbulo es que para valores bajos de Eb/No el BER es similar a la

sincronización perfecta. La técnica con peor rendimiento para este caso es la técnica de prefijo cíclico.

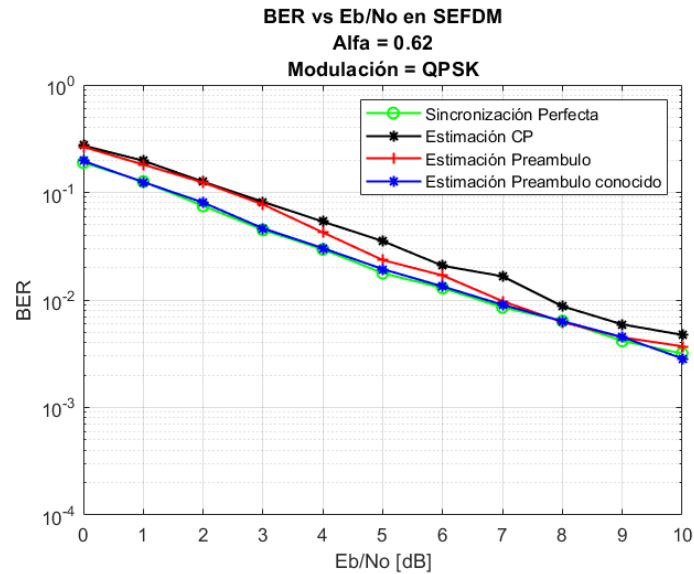


Figura 3.2. Sistema SEFDM con las 3 técnicas de estimación, alfa=0.62, QPSK.

Tabla 3.1 Valor de BER para un valor específico de Eb/No, alfa=0.62, QPSK

Eb/No(dB)	BER			
	Sincronización Perfecta	Técnica CP	Técnica Preámbulo	Técnica Preámbulo Conocido
4 muestras STO				
3	0.0448167	0.0816458	0.07735	0.0463375
7	0.0085375	0.0166042	0.00975417	0.00901667

Al realizar la simulación para el mismo factor de compresión con una modulación 16 QAM, su resultado no es el adecuado como se observa en la Figura 3.3 ya que sus valores de BER son muy altos. Lo que indica este resultado es que se debe incrementar el factor de compresión, es decir reducir el número de ceros que agrega el sistema SEFDM. Al tener un alfa=0.89 los valores BER son aceptables ya que con el aumento de Eb/No dicho valor se reduce como se observa en la Figura 3.4. También se puede recalcar que el BER obtenido para la modulación 16QAM es similar para las diferentes técnicas lo cual no sucede en la modulación QPSK que la diferencia de BER entre técnicas es mayor. En

posteriores gráficas se utilizará la configuración de factor de compresión de 0.89 para la modulación 16QAM.

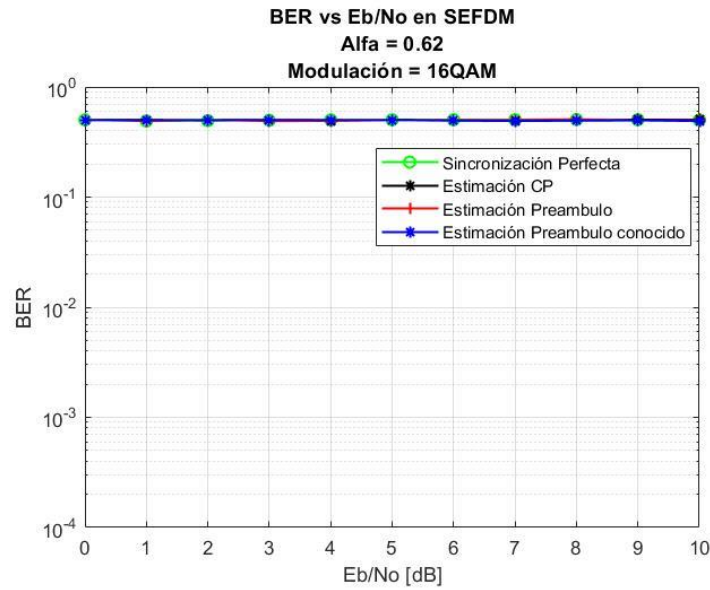


Figura 3.3. Sistema SEFDM con las 3 técnicas de estimación, alfa=0.89, 16QAM.

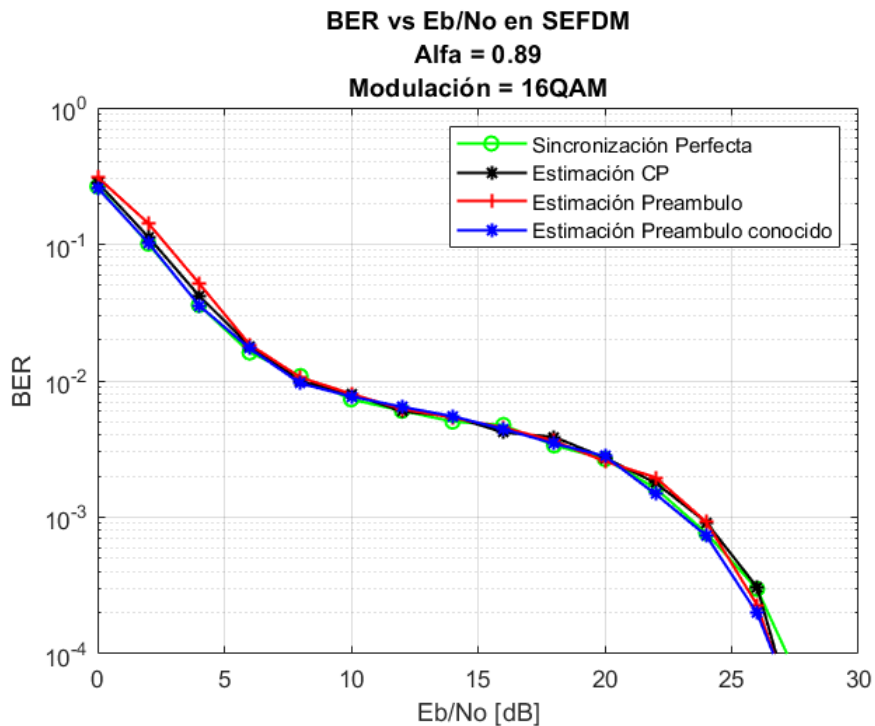


Figura 3.4. Sistema SEFDM con las 3 técnicas de estimación, alfa=0.89, 16QAM.

En la siguiente prueba de funcionamiento el valor de desplazamiento en tiempo STO es variable en el rango mencionado en el escenario de simulación, el factor de compresión se mantiene fijo en 0.89 y la modulación de 16QAM. La finalidad de obtener estos resultados

es para demostrar que las técnicas de estimación funcionan para diferentes valores de STO positivos y negativos.

En la Figura 3.5 se observa los valores obtenidos para la técnica de estimación con prefijo cíclico. La gráfica permite observar que el valor de BER no mejora ni empeora para los diferentes valores de STO positivos y negativos; por lo que se puede decir que en el rango analizado se puede utilizar cualquier valor para las siguientes simulaciones.

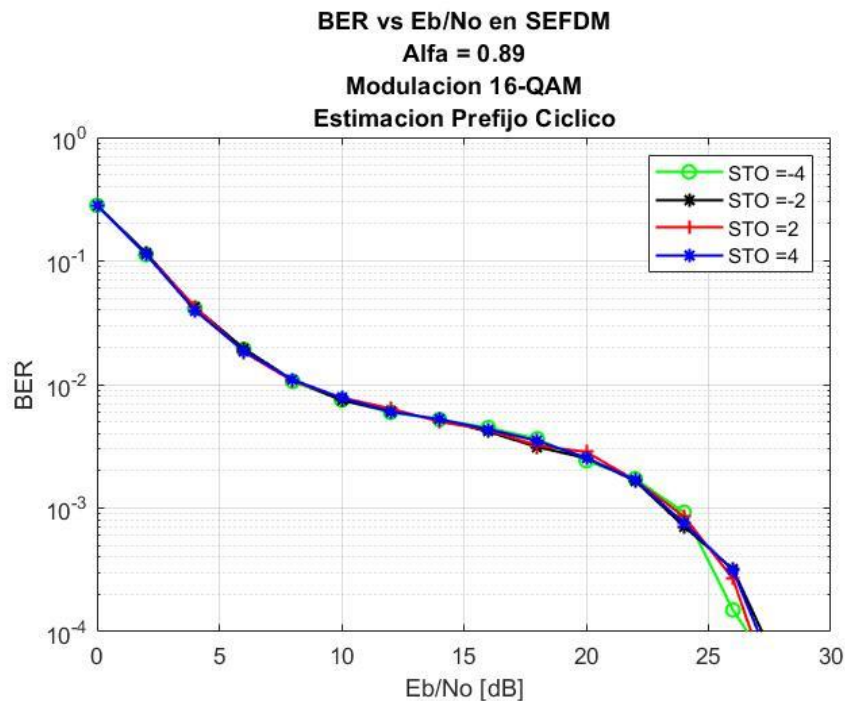


Figura 3.5. Técnica de Estimación Prefijo Cíclico con STO variable, alfa=0.89, 16QAM.

En la Figura 3.6 se observa el BER cuando se usa la técnica de estimación de preámbulo y se varía el valor de STO. A pesar de que se varía el valor de STO en el rango mencionado no se tiene una variación considerable en el BER. Es por esto que se define un valor fijo de STO para posteriores gráficas.

Finalmente, en la Figura 3.7 se tiene la técnica con el preámbulo conocido que también se ha variado los diferentes valores de STO. Las curvas para los diferentes valores de STO son similares y adecuadas para poder definir un valor fijo en las simulaciones de un escenario posterior.

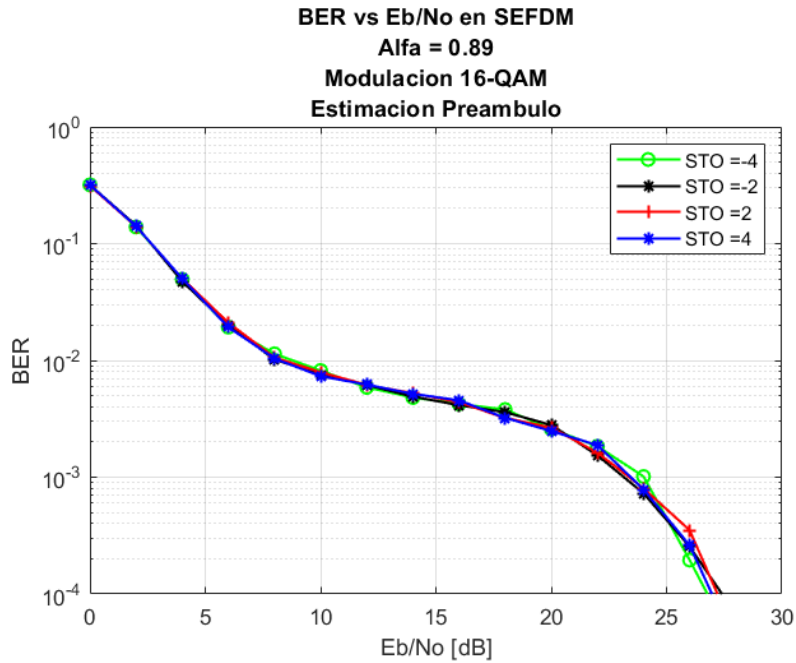


Figura 3.6. Técnica de Estimación Preámbulo con STO variable, alfa=0.89, 16QAM.

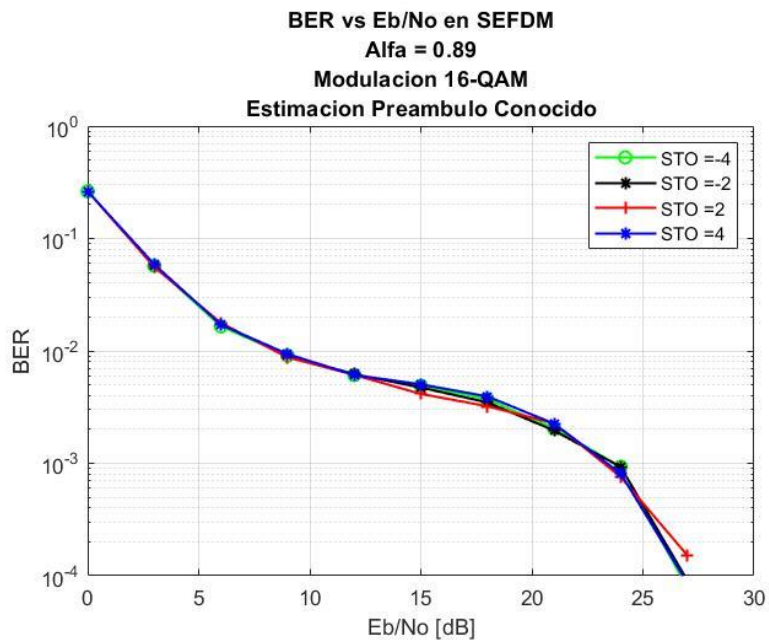


Figura 3.7. Técnica de Estimación Preámbulo Conocido con STO variable, alfa=0.89, 16QAM.

En la Figura 3.8 y 3.9 se observa las curvas BER vs Eb/No para las tres técnicas de estimación para la modulación 16QAM. El valor de alfa que se utiliza es de 0.94, 0.89 y 0.84, el desplazamiento en tiempo se fija en 4. El objetivo de esta prueba es determinar el desempeño de las técnicas con diferentes factores de compresión de ancho de banda. Para mejor apreciación de la leyenda se coloca abreviaturas donde la Técnica 1, Técnica

2 y Técnica 3, representan a la Estimación con prefijo cíclico, preámbulo y preámbulo conocido, respectivamente.

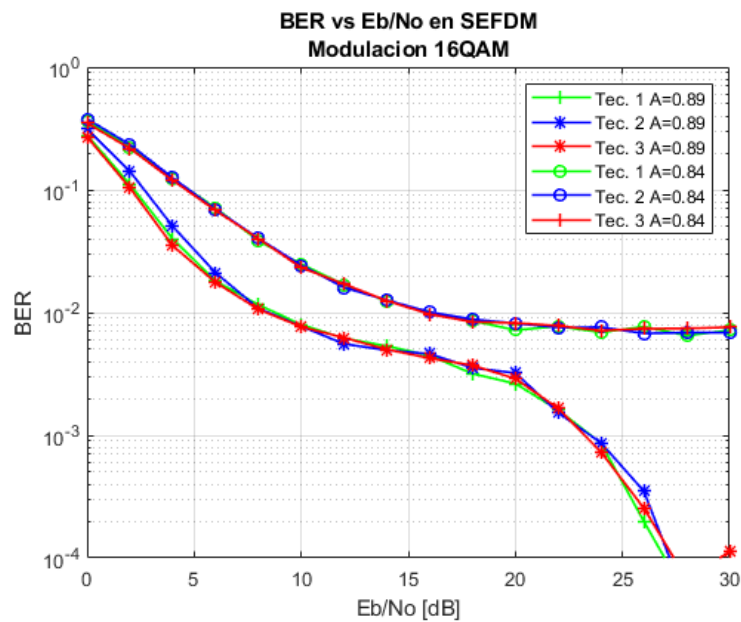


Figura 3.8. Técnicas de Estimación con alfa variable, STO=4, 16QAM.

En la modulación 16QAM para un alfa de 0.89 se obtienen valores de BER similares en las tres técnicas. Sin embargo, se observa que para un alfa de 0.84 los valores de BER son inadecuados y el rendimiento de las tres técnicas se degrada por el efecto de la compresión del ancho de banda.

En la tabla 3.3 se detalla los valores de Eb/No obtenidos para un valor fijo de BER usando un alfa de 0.94 como se observa en la Figura 3.9. El objetivo de este escenario específico es de poder comparar con la modulación 64QAM en análisis posteriores.

Tabla 3.2 Valor de BER para un valor específico de BER, STO=4, 16QAM

BER (4 muestras STO, Alfa 0.94)	Eb/No(dB)		
	Técnica CP (Tec. 1)	Técnica Preámbulo (Tec. 2)	Técnica Preámbulo Conocido (Tec. 3)
10^{-1}	1.2	2	0.8
10^{-2}	4.2	5.1	3.8
10^{-3}	7.5	7.3	5.6

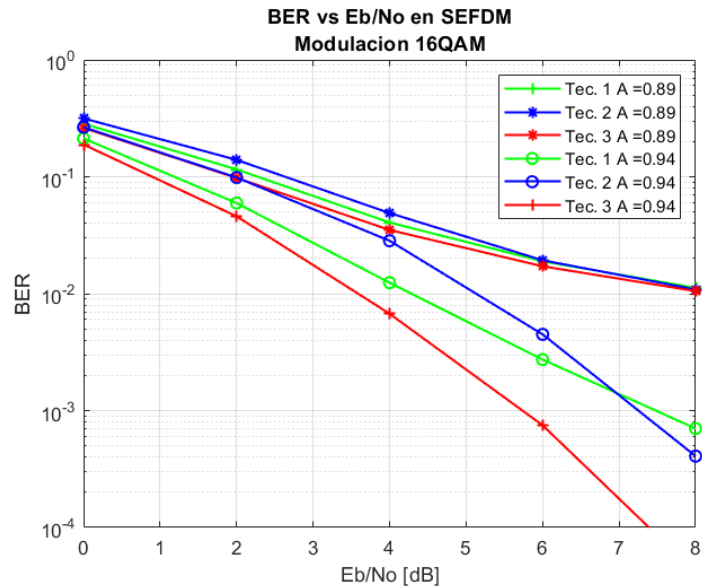


Figura 3.9. Técnicas de Estimación con alfa variable, STO=4, 16QAM.

Para una mejor apreciación de las diferentes técnicas se reduce el rango de Eb/No. Para un alfa de 0.94 se tiene diferencias considerables en el BER. Al analizar en un Eb/No fijo de 5 dB la técnica con mejor rendimiento es la del preámbulo conocido y la que tiene un peor rendimiento es la del preámbulo. Sin embargo, para un número mayor de Eb/No en 7 dB se observa que la técnica del preámbulo mejora levemente con una diferencia de 0.2 dB comparado con la del prefijo cíclico.

En la Figura 3.10 se realiza el mismo escenario anterior para la modulación 64QAM, y se mantiene el valor fijo de 4 para STO. El objetivo de esta prueba es analizar para un índice de modulación más alto el comportamiento de las tres técnicas con alfas variables de 0.89 y 0.94.

Al utilizar la modulación 64QAM se puede apreciar que para un alfa de 0.89 los valores de BER en las tres técnicas es similar. Pero al utilizar un alfa de 0.94 se puede apreciar que en el mismo rango analizado para 16QAM los valores de BER son similares es decir la diferencia de Eb/No se reduce entre cada técnica de estimación. En la tabla 3.3 se detallan los valores de Eb/No obtenidos para la modulación 64QAM con un valor de BER fijo.

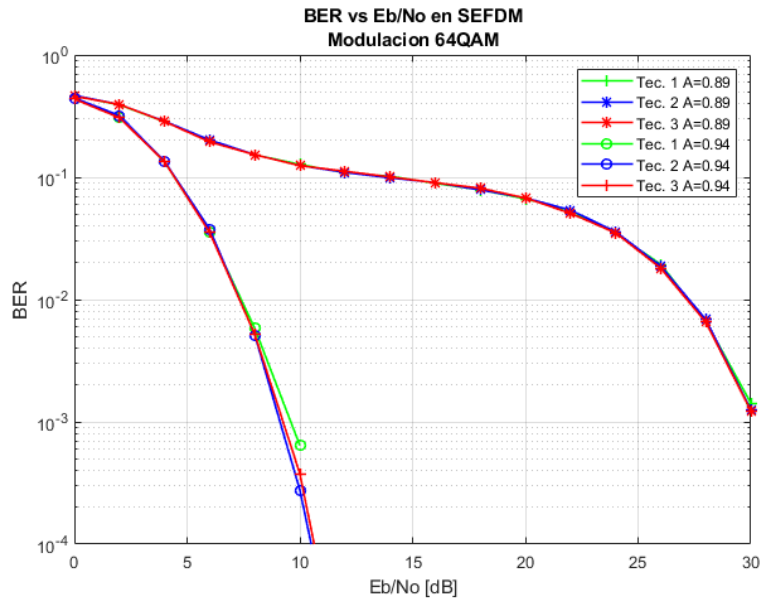


Figura 3.10. Técnicas de Estimación con alfa variable, STO=4, 64QAM.

Tabla 3.3 Valor de BER para un valor específico de BER, STO=4, 64QAM

BER (4 muestras STO, Alfa 0.94)	Eb/No(dB)		
	Técnica CP (Tec. 1)	Técnica Preámbulo (Tec. 2)	Técnica Preámbulo Conocido (Tec. 3)
10^{-1}	4.5	4.5	4.5
10^{-2}	7.5	7.5	7.5
10^{-3}	9.5	9.1	9.3

3.4 CONCLUSIONES

La sincronización en el receptor del sistema SEFDM permite demostrar como el efecto del desplazamiento en tiempo degrada la señal y, para que esta degradación se minimice es necesario implementar técnicas de estimación que permitan identificar el inicio del símbolo y así poder realizar la sincronización de forma correcta.

La secuencia de entrenamiento o preámbulo es base para implementar dos técnicas de estimación es por ello que su estructura debe estar bien definida y explicada a detalle, para luego al implementar la técnica de estimación saber los valores de ventana a utilizar.

Para la modulación QPSK con un factor de compresión de 0.62 la técnica que presenta mejores resultados es la del preámbulo conocido y la que tiene un peor rendimiento es la técnica del prefijo cíclico. Para 64QAM se obtiene valores similares del BER y se podría decir que en ese tipo de modulación el rendimiento de las tres técnicas es similar. Para el caso de 16QAM y un alfa de 0.94 la mejor técnica es la del preámbulo conocido y la que tiene un peor rendimiento es la del preámbulo en un rango de 0 a 8 dB de E_b/N_0 como se presentó en la Figura 3.9.

Al momento que se implementa la técnica de prefijo cíclico se evita el preámbulo ya que al solo basarse en el prefijo cíclico es innecesario que la técnica compare el preámbulo. Se podría decir que al comparar menor información en las ventanas debería tener un mejor rendimiento. Sin embargo, esta técnica obtuvo el peor rendimiento para un escenario con modulación QPSK y alfa de 0.62.

Al utilizar las diferentes modulaciones hay que tener en cuenta el factor de compresión. Por ejemplo, para QPSK se obtiene un rendimiento comparable entre técnicas con un alfa de 0.62, para 16QAM es de 0.89 y 64QAM es de 0.94. Los valores mencionados para cada modulación permiten que las técnicas de estimación sean evaluadas de forma correcta y con un BER aceptable.

En la técnica de estimación del preámbulo al compararse ventanas de la señal que han sido afectadas por el ruido del canal, la diferencia utilizada en el algoritmo puede contener más errores al estimar el inicio del símbolo. Sin embargo, con la técnica del preámbulo conocido se observa que esa diferencia se reduce y permite que la técnica estime el inicio del símbolo con menos errores.

Al realizar la prueba de varios factores de compresión se concluye que para tener un menor BER se requiere un factor de compresión de ancho de banda mayor. Recordando que un factor de compresión (alfa) mayor implica poca reducción del ancho de banda requerido comparado con un sistema OFDM.

Al implementar la técnica de preámbulo conocido en el transmisor la ventana del preámbulo se mantiene fija, pero al utilizar la comparación del segundo símbolo se observa que para valores negativos de STO se tiene un BER demasiado alto es por esto que se utiliza únicamente la comparación del símbolo 1. Al realizar pruebas de esta forma se comprueba que se puede utilizar el intervalo de STO establecido para las otras técnicas.

Se comprueba que la técnica implementada en base al preámbulo conocido en el transmisor obtiene el mejor rendimiento y justifica su implementación ya que para un

sistema de comunicaciones con mayor sensibilidad este tipo de técnica sería la que brinde el mejor desempeño en la comunicación, cabe recalcar que utilizando las configuraciones adecuadas se obtienen dichos resultados.

3.5 RECOMENDACIONES

Al utilizar el canal AWGN se sabe que es lo más básico que se tiene en un canal inalámbrico para futuros trabajos se podría incluir canales de tipo Rayleigh y Rician y evaluar el efecto o degradación de BER que posee el sistema con esas configuraciones.

Se podría implementar la simulación con otro tipo de detectores en el receptor y evaluar la degradación del BER. Esto se lo plantea ya que existen otros trabajos que incluyen varios detectores como configuración inicial del programa.

Se recomienda variar el tamaño de ventana y la tasa de codificación convolucional, para poder observar como las técnicas de sincronización se comportan al tener un sistema con mayor o menor redundancia.

Se recomienda implementar la simulación de otras técnicas de estimación para poder evaluar el rendimiento y su desempeño en sistemas SEFDM. Para esto también se podría aumentar el rango de valores de STO y ver hasta qué valor se mantiene el rendimiento de las técnicas STO.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] J. Prieto Blázquez, "Introducción a los sistemas de comunicación inalámbricos," *Universitat Oberta de Catalunya*, pp. 5–16, 2015.
- [2] I. Darwazeh, H. Ghannam, and T. Xu, "The First 15 Years of SEFDM: A Brief Survey," *2018 11th International Symposium on Communication Systems, Networks and Digital Signal Processing, CSNDSP 2018*, Sep. 2018, doi: 10.1109/CSNDSP.2018.8471886.
- [3] A. Dutta, K. Vinoth Babu, G. Ramachandra Reddy, Shivangi, and B. K. Nallagatla, "Symbol timing offset (STO) correction in orthogonal frequency division multiplexing (OFDM) based systems," *International Conference on Communication and Signal Processing, ICCSP 2014 - Proceedings*, pp. 508–512, Nov. 2014, doi: 10.1109/ICCSP.2014.6949894.
- [4] M. Rodrigues and I. Darwazeh, "A Spectrally Efficient Frequency Division Multiplexing Based Communication System," *8th International OFDM-Workshop (InOWo'03)*, pp. 70–74, 2003.
- [5] I. Kanaras, A. Chorti, M. R. D. Rodrigues, and I. Darwazeh, "Spectrally efficient FDM signals: Bandwidth gain at the expense of receiver complexity," *IEEE International Conference on Communications*, 2009, doi: 10.1109/ICC.2009.5199477.
- [6] T. Xu and I. Darwazeh, "Spectrally efficient FDM: Spectrum saving technique for 5G?," *Proceedings of the 2014 1st International Conference on 5G for Ubiquitous Connectivity, 5GU 2014*, pp. 273–278, Feb. 2014, doi: 10.4108/ICST.5GU.2014.258120.
- [7] S. M. Pupiales Angamarca, "Análisis de ganancia de ancho de banda y degradación de la BER en un sistema SEFDM con respecto a OFDM usando MATLAB," Escuela Politécnica Nacional, Quito, 2020.
- [8] Y. Soo. Cho, Jaekwon. Kim, and W. Young. Yang, "MIMO-OFDM Wireless Communications with MATLAB.," p. 457, 2010.
- [9] W. Aziz and S. Saleem, "Time Offset Estimation for OFDM Using MATLAB ," *Journal of Expert Systems (JES)*, vol. 1, pp. 59–61, 2012.
- [10] G. Logacho, "Simulación en MATLAB de los detectores ZF, MMSE e ID para un sistema SEFDM utilizando un canal con desvanecimiento.," Escuela Politécnica Nacional, Quito, 2021.

5 ANEXOS

ANEXO I

Código que contiene el script y funciones en formato (.m) de la simulación en MATLAB
(Anexo Digital)