



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Algoritmos y arquitecturas hardware para la implementación de OFDM en sistemas de comunicaciones ópticos

Noviembre de 2018

Autor: Julián Santiago Bruno

Director: Dr. Vicenç Almenar Terré
Dr. Javier Valls Coquillat

*A mi amada esposa Natalia
y a nuestras dos maravillosas hijas, Catalina y Olivia.*

Resumen

Esta tesis explora en profundidad la viabilidad técnica y las prestaciones de un sistema de transmisión para comunicaciones ópticas, de bajo coste y alta velocidad, basado en la multiplexación por división de frecuencia ortogonal (OFDM) mediante la implementación de los algoritmos de procesamiento digital de la señal en tiempo real sobre dispositivos de lógica programable (FPGA). Los sistemas de transmisión de modulación de intensidad y detección directa (IM/DD) parecen ser la solución más interesante para las redes ópticas pasivas (PONs) de bajo coste que serán necesarias para responder a la gran demanda de tráfico de los próximos años, producto del aumento significativo de dispositivos conectados a internet, servicios y programas en la nube, vídeo de alta definición, entre otros.

Por tanto, esta tesis tiene como objetivo principal obtener la máxima tasa binaria y eficiencia espectral posible de un sistema IM/DD OFDM en PON (de una sola banda y una sola longitud de onda). Con esta finalidad se ha desarrollado la arquitectura *hardware* de un receptor OFDM de alta velocidad que opera en tiempo real en un dispositivo FPGA Virtex-7 a una frecuencia de reloj de 312,5 MHz utilizando un conversor analógico digital con una tasa de muestreo de 5 GS/s. Para lograr las mejores prestaciones posibles, se ha intentado aprovechar al máximo el ancho de banda del sistema (acercándose al límite de Nyquist) y se ha realizado una carga variable de las subportadoras del símbolo OFDM atendiendo a las características del canal electro-óptico. Además, se han diseñado e implementado los algoritmos de procesamiento necesarios para la detección y demodulación de los símbolos OFDM, y se ha desarrollado una plataforma experimental que ha permitido validarlos en tiempo real a través de

un enlace de fibra monomodo estándar (SSMF). El principal resultado de esta tesis es el haber demostrado experimentalmente que con el sistema propuesto se puede alcanzar una tasa binaria de 19,63 Gb/s y una eficiencia espectral de 8,07 bit/s/Hz sobre 20 km de SSMF, lo cual implica casi duplicar las mejores prestaciones publicadas hasta la fecha.

Para completar el diseño del receptor, se ha desarrollado un algoritmo de sincronización de tiempo basado en la correlación cruzada entre un preámbulo conocido y la señal OFDM recibida, el cual presenta una tasa de error extremadamente baja, aun en condiciones de muy baja SNR, y su diseño optimizado requiere menos recursos que otros sincronizadores publicados hasta la fecha. También se ha desarrollado un procesador de FFT de longitud variable mediante una arquitectura paralela segmentada que permite alcanzar hasta una tasa de 10 GS/s en dispositivos Virtex-7 con una eficiencia (área/velocidad) superior a la de otros trabajos publicados, y un ecualizador de canal basado en una técnica de estimación y compensación en frecuencia, que utiliza un preámbulo característico para poder disminuir la complejidad *hardware* y aumentar la precisión de la estimación. Todos los algoritmos implementados en esta tesis doctoral han sido diseñados para procesar 16 muestras en paralelo, y así poder reducir la frecuencia de reloj (5 GHz/16) hasta valores aceptables para los dispositivos FPGA.

Palabras Claves: *TSA, FFT, OFDM, SMF, IM/DD, FPGA y alta velocidad.*

Resum

Aquesta tesi explora en profunditat la viabilitat tècnica i les prestacions d'un sistema de transmissió per a comunicacions òptiques, de baix cost i alta velocitat, basat en la multiplexació per divisió en freqüència ortogonal (OFDM) mitjançant la implementació d'algorismes de processat digital del senyal en temps real sobre dispositius de lògica programable (FPGA). Els sistemes de transmissió de modulació d'intensitat i detecció directa (IM/DD) semblen ser la solució més interessant per a les xarxes òptiques passives (PON) de baix cost que seran necessàries per respondre a la gran demanda de tràfic dels propers anys, producte de l'augment significatiu de dispositius connectats a Internet, serveis i programari al núvol, vídeo d'alta definició, entre d'altres.

Per tant, aquesta tesi té com objectiu principal obtenir la màxima taxa binària i eficiència espectral possible d'un sistema IM/DD OFDM en PON (amb una sola banda i una sola longitud d'ona). Amb aquesta finalitat s'ha desenvolupat l'arquitectura hardware d'un receptor OFDM d'alta velocitat que treballa en temps real a un dispositiu FPGA Virtex-7 amb una freqüència de rellotge de 312,5 MHz utilitzant un convertidor analògic a digital amb una taxa de mostreig de 5 GS/s. Per aconseguir les millors prestacions possibles, s'ha intentat aprofitar al màxim l'ample de banda del sistema (aproximant-se al límit de Nyquist) i s'ha realitzat una càrrega variable de les subportadores del símbol OFDM atenent a les característiques del canal electrò-òptic. A més, s'han dissenyat e implementat els algorismes de processament necessaris per la detecció i desmodulació dels símbols OFDM, i s'ha desenvolupat una plataforma experimental que ha permès validar-los en temps real a través d'un enllaç de fibra monomode estàndard (SSMF). El principal resultat d'aquesta

tesi és haver demostrat experimentalment que amb el sistema proposat es pot arribar a una taxa binària de 19,63 Gb/s amb una eficiència espectral de 8,07 bits/s/Hz sobre 20 km de SSMF, la qual cosa implica quasi duplicar les millors prestacions publicades fins aquest moment.

Per completar el disseny del receptor, s'ha desenvolupat un algorisme de sincronització de temps basat en la correlació creuada entre un preàmbul conegut i el senyal OFDM rebut, el qual presenta una taxa d'error extremadament baixa, inclòs en condicions de molt baixa SNR, i el seu disseny optimitzat requereix menys recursos que altres sincronitzadors publicats fins el moment. També s'ha desenvolupat un processador de FFT de longitud variable mitjançant una arquitectura paral·lela segmentada que permet arribar fins una taxa de 10 GS/s en dispositius Virtex-7 amb una eficiència (àrea/velocitat) superior a la d'altres treballs publicats, i un equalitzador de canal basat en una tècnica d'estimació i compensació en freqüència, que utilitza un preàmbul característic per poder disminuir la complexitat hardware i augmentar la precisió de l'estimació. Tots els algorismes implementats a aquesta tesi doctoral han sigut dissenyats per processar 16 mostres en paral·lel, i així poder reduir la freqüència de rellotge (5 GHz/16) fins valors acceptables pels dispositius FPGA.

Paraules Claus: TSA, FFT, OFDM, SMF, IM/DD, FPGA i alta velocitat.

Abstract

This thesis presents an in-depth exploration of the technical feasibility and achievable performance of a low-cost and high-speed optical communication system based on orthogonal frequency division multiplexing (OFDM) through the implementation of real-time digital signal processing algorithms over programmable logic devices (FPGA). Optical transmission systems based on intensity modulation and direct detection (IM/DD) is considered as one of the most interesting solutions for the deployment of the low-cost passive optical networks (PONs) that will be needed to cover the high traffic demand in the coming years. This demand is fueled, among others, by the significant increase of connected devices to the Internet, services and programs in the cloud, high definition video, etc.

The main objective of this thesis is to achieve the maximum bitrate and spectral efficiency of an IM/DD PON OFDM system (using a single band and a single wavelength). To this end, the hardware architecture of a high-speed real-time OFDM receiver, including all the necessary algorithms to perform the detection and demodulation of the OFDM symbols, has been implemented in a Virtex-7 FPGA device at a clock frequency of 312.5 MHz using a digital analog converter with a sampling rate of 5 GS/s. To reach the best possible performance, all the system bandwidth has been employed and the OFDM sub-carriers have been loaded according to the characteristics of the electro-optical channel. An experimental platform for optical transmission through standard single-mode fiber (SSMF) has been developed to evaluate in real-time the performance of the implemented receiver. The main result of this thesis is the experimental validation of the proposed system that has achieved a bit rate of

19.63 GS/s and a spectral efficiency of 8.07 bit/s/Hz over 20 km SSMF. These results almost double the best performance published to date.

The receiver implementation included the design and development of several algorithms. First, it was designed a time synchronization algorithm (TSA) based on the cross-correlation between a known preamble and the received OFDM signal. This TSA has a good performance in low-SNR scenarios and its optimized design requires fewer resources than other synchronizers published in the literature. Second, a variable length parallel pipelined FFT processor has been implemented in a Virtex-7 device, it reaches a throughput of 10 GS/s with an efficiency (area/speed) higher than that of other published works. And finally, a channel equalizer working in the frequency domain to estimate and compensate channel distortions, which uses a known preamble to decrease the hardware complexity and increase the accuracy of the estimation, has been implemented. All the algorithms in this thesis have been developed to process 16 samples in parallel, thus reducing the required clock frequency (5 GHz/16) to acceptable values for the FPGA devices.

Keywords: *TSA, FFT, OFDM, SMF, IM/DD, FPGA and high speed.*

Agradecimientos

En primer lugar, quisiera agradecer a mis directores de tesis, el Dr. Javier Valls y el Dr. Vicenç Almenar, por todo el apoyo que me han brindado, por sus excelentes consejos y por haberme compartido sus conocimientos sin ningún tapujo, no sólo durante el desarrollo de esta tesis doctoral, sino también durante la tesis del máster.

Deseo expresar mi gratitud a las autoridades de la Facultad Regional Buenos Aires de la Universidad Tecnológica Nacional por el apoyo económico y personal recibido durante la realización de esta tesis. En particular, al decano de la Facultad, Ing. Guillermo Oliveto, al secretario general, Juan Manuel Tiribelli, y al hasta hace poco director del Departamento de Electrónica, Ing. Alejandro Furfaro.

Agradecerle a mi familia por el apoyo recibido durante todos estos años, lejos de mi tierra y mis afectos. De ellos aprendí la constancia del trabajo y que todo esfuerzo, tarde o temprano, tiene su recompensa. Sin dudas ha sido un camino largo y de muchos sacrificios, el tiempo será el mejor juez y esta tesis el mejor testigo.

Por sobre todas las cosas quiero darle las gracias a mi esposa y a mis dos hijas, por su paciencia, comprensión y apoyo incondicional con este proyecto, y por el tiempo que me han concedido, un tiempo robado a nuestra vida familiar. Sin su apoyo este trabajo nunca hubiese sido posible y, por esto, este trabajo también les pertenece.

Finalmente, esta tesis doctoral se ha desarrollado en el marco de dos proyectos de investigación financiados por el Ministerio de Economía y Competitividad de España denominados:

- “Procesado Digital de Señales Ópticas en Medios Guiados (OPTOPRO)”, *TEC2012-38558-C02-02*, del 01/01/2013 al 01/01/2016.
- “Tratamiento Digital de la Señal y Corrección de Errores en Transmisión Óptica mediante Fibra Multi-Núcleo para Redes Ópticas de Acceso y de Transporte Celular (OPTODSP)”, *TEC2015-70858-C2-2-R*, del 01/01/2016 al 01/01/2019.

Índice general

Resumen	V
Resum	VII
Abstract	IX
Agradecimientos	XI
Índice general	XIII
Lista de Figuras	XXI
Lista de Tablas	XXIV
Lista de Siglas	XXV

1	Introducción	1
1.1	OFDM para comunicaciones ópticas	3
1.1.1	OFDM, origen y evolución	6
1.1.2	Principios básicos de OOFDM	7
1.2	Antecedentes y estado actual	12
1.3	Motivación y objetivos	14
1.4	Metodología de trabajo	15
1.5	Contribuciones de la tesis	16
1.6	Organización de la tesis	18
2	Caracterización de la plataforma de experimentación	21
2.1	Convertor DA	23
2.2	Amplificador eléctrico	30
2.3	Etapa óptica	31
2.4	Filtro Antisolapamiento	32
2.5	Convertor AD	33
2.5.1	Calibración analógica	37
2.6	Reloj de muestreo	41
2.7	Conclusiones	42
3	Diseño de un módem OFDM para comunicaciones ópticas	45
3.1	Transmisor OFDM	46
3.1.1	Generador Arbitrario de Funciones	46
3.1.2	Generación de trama OFDM	47
3.1.3	Optimización del Transmisor	48
3.1.4	Modelo de precisión finita	57
3.1.5	Simulación	58
3.2	Receptor OFDM	58
3.2.1	Sincronizador Temporal	59
3.2.2	Transformada Rápida de Fourier (FFT)	72
3.2.3	Ecualizador de canal	87
3.2.4	Desmapeo	94
3.2.5	Simulación	103

3.3 Conclusiones	105
4 Implementación hardware del módem OFDM óptico	107
4.1 Transmisor OFDM	108
4.1.1 Interfaz DAC	108
4.1.2 Memoria de almacenamiento	111
4.1.3 Control e Interfaz de usuario	111
4.1.4 Puesta en marcha	112
4.2 Receptor OFDM	114
4.2.1 Interfaz ADC	114
4.2.2 Sincronizador Temporal	120
4.2.3 Transformada Rápida de Fourier (FFT)	121
4.2.4 Ecuador de canal	127
4.2.5 Desmapeo	129
4.3 Resultados de implementación	130
4.4 Simulación	133
4.5 Conclusiones	134
5 Plataforma de experimentación	137
5.1 Descripción de la plataforma	138
5.2 Estimación del canal.	139
5.3 Medidas para carga constante	141
5.4 Medidas para carga variable.	144
5.5 Comparación con otros diseños.	147
5.5.1 Sistemas de alto coste	149
5.6 Análisis del desplazamiento de frecuencia de reloj de muestreo.	150
5.7 Conclusiones	152
6 Conclusiones y trabajos futuros	155
6.1 Conclusiones	155
6.2 Trabajos futuros	158
Bibliografía	159

Lista de Figuras

1.1.	Diagrama de conectividad de extremo a extremo: AN, MAN, WAN. Fuente: [3].	3
1.2.	Diagrama de bloques del primer sistema de multiplexación de frecuencia ortogonal de tres canales. Fuente: [30].	6
1.3.	Diagrama de bloques de un sistema IM/DD PON OOFDM.	8
2.1.	Banco de pruebas para caracterización del sistema IM/DD OFDM para comunicación óptico.	22
2.2.	Medición de onda reflejada para un pulso rectangular generado con el DAC.	24
2.3.	Respuesta en frecuencia del DAC MD657B operando a 5 GS/s.	25
2.4.	Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.	27
2.4.	(Continuación) Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.	28
2.4.	(Continuación) Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.	29
2.5.	Respuesta en frecuencia del amplificador del DAC.	30

2.6. Módulo de la respuesta en frecuencia de la etapa electro-óptica para 10/20/40 km de SSMF.	31
2.7. Respuesta en frecuencia del filtro antisolapamiento.	32
2.7. (Continuación) Respuesta en frecuencia del filtro antisolapamiento.	33
2.8. Espectros de los tonos capturados con el ADC EV10AQ190 operando a 5 GS/s.	36
2.9. Histogramas de los 4 canales con sus entradas a 0 V sin calibración de <i>offset</i>	38
2.10. Histogramas de los 4 canales con sus entradas a 0 V con calibración de <i>offset</i>	38
2.11. Espectro para un tono puro de 600 MHz con y sin calibración de <i>offset</i>	39
2.12. Espectro para un tono puro de 600 MHz con y sin calibración de ganancia.	40
2.13. Generación de relojes de muestreo para DAC y ADC.	42
2.14. Espectro de frecuencia de los relojes de muestreo.	43
3.1. Estructura de la trama OFDM propuesta.	47
3.2. Efecto de la relación de recorte para diferentes bits de cuantificación (DAC+ADC).	50
3.3. Efecto de la relación de recorte utilizando 9 bits DAC y 7 bits ADC.	50
3.4. Caracterización del módem OFDM para el caso de EBTB.	51
3.5. BER vs EVM para monoportadora y el marcador para una BER de $3,8 \times 10^{-3}$	52
3.6. Respuesta en frecuencia de un conversor digital analógico ideal.	53
3.7. Respuesta en frecuencia del DAC en modo NRZ y de los posibles preénfasis.	55
3.8. Señal OFDM sin preénfasis digital para un esquema EBTB.	56

3.9. Señal OFDM con preénfasis digital para un esquema EBTB.	57
3.10. Diagrama de bloques DSP del receptor OFDM.	58
3.11. Diagrama de bloques del algoritmo de sincronización de tiempo propuesto.	60
3.12. Preámbulo propuesto.	61
3.13. Diagrama de bloques del correlador cruzado.	62
3.14. Magnitud de la salida del correlador cruzado ($P[n]$) y filtro ($M[n]$) durante la parte repetitiva del preámbulo y en el co- mienzo del intervalo de guarda.	63
3.15. Diagrama de bloques del TSA paralelo.	65
3.16. Curvas de PCTD para el TSA propuesto ($N_{ss}=32$) versus SNR para diferentes esquemas de cuantificación de la señal recibida y el marcador para una BER de 10^{-2}	67
3.17. Curvas de PCTD versus ROP para diferentes esquemas de me- dición y el marcador para una BER de 10^{-2}	71
3.18. Curvas de BER versus ROP para diferentes esquemas de medi- ción y el marcador para una BER de 10^{-2}	71
3.19. Arquitectura MDC PFFFT con $O=4$ y $R=4$ de n etapas.	74
3.20. Elemento computacional <i>radix</i> -4 con decimación en frecuencia.	78
3.21. Conmutador de demoras de la etapa k -ésima de la FFT segmen- tada con <i>radix</i> - R	79
3.22. Patrones de datos a través del conmutador de demoras de $R=4$ con $f=4$	81
3.23. Arquitectura de longitud variable R4MDC PFFFT con 4 cami- nos de datos. S_k simboliza las etapas FFT, incluidos los bloques CEs y DCs.	82
3.24. Patrones de datos de entrada/salida a través de la arquitectura R4MDC PFFFT 64 puntos con 4 caminos de datos ($N=64$ y $P=16$).	84

3.25. Curva de BER vs SNR para señal OFDM modulada con 512-QAM utilizando una FFT de 1024 puntos con 10 bits de entrada, 20 bits de salida y 18 bits de TF.	87
3.26. BER vs SNR para diferentes número de diezmado y repeticiones, para 512 subportadoras moduladas con 256 QAM.	93
3.27. Constelaciones 16-QAM, 32-QAM, 64-QAM y 128-QAM con mapeo Gray.	94
3.28. Constelaciones 256-QAM y 512-QAM con mapeo Gray.	95
3.29. Regiones de desmapeo para el caso de 16-QAM.	97
3.30. Regiones de desmapeo para el caso de 32-QAM.	99
3.31. Diagramas de constelación 16-QAM para distintos tipos de señales utilizando algoritmos sin y con precisión finita. (a) y (d) tipo I (EVM=0,45 %), (b) y (e) tipo II (EVM=12 %), y (c) y (f) tipo III (EVM=2,9 %).	104
4.1. Diagrama de bloques de la interfaz con el DAC.	109
4.2. Diagrama de formas de onda para la selección del muestreo óptimo de los bits de las muestras del conversor MD657B.	110
4.3. Banco de pruebas para verificación del control y generación del DAC.	112
4.4. Pruebas de selección de fase del conversor DA con un tono puro de 366 MHz.	113
4.5. Diagrama de bloques del conversor EV10AQ190A.	115
4.6. Diagrama de bloques de la interfaz con el ADC.	116
4.7. Algoritmo de alineamiento de bit de la interfaz del ADC.	117
4.8. Banco de pruebas para verificación del control y adquisición del ADC.	119
4.9. Multiplicador de números complejos, X corresponde a valores intermedios de la FFT e Y corresponde a valores de factores de rotación.	123

4.10. Implementación del <i>hardware</i> de reordenamiento de las muestras de entrada/salida.	126
4.11. Diagrama de bloques del estimador de canal.	128
4.12. Diagrama de bloques del compensador de canal.	129
5.1. Configuración experimental para el sistema IM/DD OOFDM en tiempo real.	138
5.2. Estimación del canal para diferentes esquemas de transmisión. .	140
5.3. EVM por subportadora para diferentes esquemas de transmisión.	140
5.4. Diagramas de constelaciones para una transmisión sobre 10 km de fibra utilizando diferentes órdenes de modulación QAM. . .	142
5.5. Cantidad de errores y BER por subportadora para una transmisión sobre 10 km de fibra utilizando una modulación 256-QAM.	144
5.6. Diagramas de constelación para diferentes subconjuntos de subportadoras para una transmisión sobre 10 km de fibra utilizando una modulación 256-QAM.	145
5.7. Carga binaria por subportadora para diferentes esquemas de transmisión.	146

Lista de Tablas

3.1. Comparación de complejidad de TSAs	68
3.2. Valores de Q de las diferentes etapas de una FFT de N puntos	78
3.3. Valores de f para diferentes longitudes de FFT con $R=4$	80
3.4. Orden natural para PPFFT con $P=16$	83
3.5. Orden de entrada para PPFFT con $P=16$	83
3.6. Complejidad <i>hardware</i> del desmapeador M-QAM	101
4.1. Transmisor OOFDM - Recursos utilizados de la FPGA	131
4.2. Receptor OOFDM - Recursos utilizados de la FPGA	131
4.3. Comparación de características y recursos utilizados de la FPGA por receptores OOFDM	132
5.1. Prestaciones del sistema propuesto utilizando diferentes modu- laciones: 64-, 128-, 256- y 512-QAM.	142
5.2. Prestaciones del sistema propuesto utilizando <i>bit-loading</i>	147
5.3. Comparación de sistemas IM/DD OOFDM	147

5.4. Mediciones de EVM para diferentes valores de SCFO 152

Lista de Siglas

ACO-OFDM	<i>Asymmetrically Clipped Optical OFDM</i>
ADC	<i>Analog-to-Digital Converter</i>
ADO-OFDM	<i>Asymmetrically clipped DC biased Optical OFDM</i>
ADSL	<i>Asymmetric Digital Subscriber Line</i>
AGC	<i>Automatic Gain Control</i>
AMOOFDM	<i>Adaptively Modulated OOFDM</i>
AN	<i>Access Network</i>
ASIC	<i>Application-Specific Integrated Circuit</i>
AWG	<i>Arbitrary Waveform Generator</i>
AWGN	<i>Additive White Gaussian Noise</i>
BER	<i>Bit Error Rate</i>
BL	<i>Bit Loading</i>
BPSK	<i>Binary Phase Shift Keying</i>
BRAM	<i>Block Random-Access Memory</i>
BW	<i>Bandwidth</i>
CE	<i>Computational Element</i>

CD	<i>Chromatic Dispersion</i>
CFO	<i>Carrier Frequency Offset</i>
CIU	<i>Control e Interfaz de Usuario</i>
CO-OFDM	<i>Coherent Optical OFDM</i>
CP	<i>Cyclic Prefix</i>
DAC	<i>Digital-to-Analog Converter</i>
DC	<i>Delay Commutator</i>
DCO-OFDM	<i>DC biased Optical OFDM</i>
DDR	<i>Double-Data-Rate</i>
DFB	<i>Distributed Feedback Laser</i>
DFT	<i>Discrete Fourier Transform</i>
DIT	<i>Decimation-In-Time</i>
DIF	<i>Decimation-In-Frequency</i>
DML	<i>Directly Modulated Laser</i>
DSO	<i>Digital Sampling Oscilloscope</i>
DVB-H	<i>Digital Video Broadcasting - Handheld</i>
DVB-SH	<i>Digital Video Broadcasting - Services to Handheld</i>
DVB-T	<i>Digital Video Broadcasting - Terrestrial</i>
DVB-T2	<i>Digital Video Broadcasting - Terrestrial 2</i>
EAM	<i>Electro-Absortion Modulator</i>
EBTB	<i>Electrical Back-To-Back</i>
EC	<i>Computational Element</i>
EDC	<i>Electrical Dispersion Compensation</i>
ENOB	<i>Effective Number Of Bits</i>
EVM	<i>Error Vector Magnitude</i>

FDM	<i>Frequency-Division Multiplexing</i>
FEC	<i>Forward Error Correction</i>
FF	<i>Flip-Flop</i>
FFT	<i>Fast Fourier Transform</i>
FPGA	<i>Field Programmable Gate Array</i>
GI	<i>Guard Interval</i>
HD-FEC	<i>Hard-Decision Forward Error Correction</i>
HDL	<i>Hardware Description Language</i>
HUE	<i>Hardware Utilization Efficiency</i>
ICI	<i>Inter Carrier Interference</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IFFT	<i>Inverse Fast Fourier Transform</i>
ILA	<i>Integrated Logic Analyzer</i>
IM/DD	<i>Intensity Modulation and Direct Detection</i>
INL	<i>Integral Nonlinearity</i>
ISI	<i>Inter Symbol Interference</i>
LPF	<i>Low Pass Filter</i>
LR	<i>Long-Reach</i>
LS	<i>Long Symbol</i>
LSB	<i>Low Significant Bit</i>
LTE	<i>Long Term Evolution</i>
LUT	<i>Look-Up Table</i>
LVDS	<i>Low Voltage Differential Signaling</i>
MAN	<i>Metropolitan-Area Network</i>
MDC	<i>Multi-path Delay Commutator</i>

MIMO	<i>Multiple-Input and Multiple-Output</i>
MSE	<i>Minimum Square Error</i>
MUX	<i>Multiplexor</i>
MZM	<i>Mach-Zehnder Modulator</i>
NH	<i>Normal Hold</i>
NRZ	<i>Non-Return-to-Zero</i>
OAN	<i>Optical Access Network</i>
OBTB	<i>Optical Back-To-Back</i>
ODV	<i>Output Data Valid</i>
OFDM	<i>Orthogonal Frequency Division Multiplexing</i>
OOFD	<i>Optical OFDM</i>
OSV	<i>Output Synchronization Valid</i>
PAPR	<i>Peak-to-Average Power Ratio</i>
PCTD	<i>Probability of Correct Time Detection</i>
PD	<i>PhotoDiode</i>
PPFT	<i>Pipelined Fast Fourier Transform</i>
PMD	<i>Polarization-Mode Dispersion</i>
PON	<i>Passive Optical Network</i>
PPFFT	<i>Parallel Pipelined Fast Fourier Transform</i>
QAM	<i>Quadrature Amplitude Modulation</i>
QPSK	<i>Quadrature Phase-Shift Keying</i>
R2MDC	<i>Radix-2 Multi-Delay Commutator</i>
R4MDC	<i>Radix-4 Multi-Delay Commutator</i>
RAM	<i>Random-Access Memory</i>
RDADDR	<i>Read Address</i>

ROM	<i>Read-Only Memory</i>
ROP	<i>Received Optical Power</i>
RT	<i>Real-Time</i>
RZ	<i>Return-to-Zero</i>
SCFO	<i>Sampling Clock Frequency Offset</i>
SDC	<i>Single-path Delay Commutator</i>
SDF	<i>Simple-path Delay Feedback</i>
SD-FEC	<i>Soft-Decision Forward Error Correction</i>
SDP	<i>Simple Dual-Port</i>
SE	<i>Spectral Efficiency</i>
SFDR	<i>Spurious-Free Dynamic Range</i>
SINAD	<i>Signal to Noise And Distortion ratio</i>
SISO	<i>Single-Input and Single-Output</i>
SMF	<i>Single-Mode Fiber</i>
SNR	<i>Signal-to-Noise Ratio</i>
SPI	<i>Serial Peripheral Interface</i>
SR	<i>Short-Reach</i>
SRL	<i>Shift Register LUT</i>
SS	<i>Short Symbol</i>
SSB	<i>Single-Side-Band</i>
SSBI	<i>Signal-Signal Beat Interference</i>
SSII	<i>Subcarrier-to-Subcarrier Intermixing Interference</i>
SSMF	<i>Standard Single-Mode Fiber</i>
TDM	<i>Time Division Multiplexing</i>
TF	<i>Twiddle Factor</i>

TI	<i>Time-Interleaved</i>
TS	<i>Training Sequence</i>
TSA	<i>Time Synchronization Algorithm</i>
VDSL2	<i>Very high speed Digital Subscriber Line 2</i>
VHDL	<i>Very High Speed Integrated Circuit HDL</i>
VLSI	<i>Very Large Scale Integration</i>
WAN	<i>Wide-Area Network</i>
WDM	<i>Wavelength Division Multiplexing</i>
WiMAX	<i>Worldwide Interoperability for Microwaves Access</i>
WLAN	<i>Wireless Local Area Network</i>
WRADDR	<i>Write Address</i>
ZFE	<i>Zero-Forcing Equalization</i>

Capítulo 1

Introducción

Durante más de 30 años de desarrollo, las redes de comunicación han atravesado tres grandes fases: voz, Internet y vídeo. Impulsadas por la interconexión de centros de datos en la nube (*DC cloud*), los servicios de vídeo de ultra ancho de banda y los servicios 5G, las redes de comunicación óptica evolucionarán hacia redes simplificadas en el futuro. El tráfico de vídeo 4K/8K y realidad aumentada/virtual son sensibles a la latencia de la red, las redes financieras y de automatización industrial requieren alta seguridad, y las líneas dedicadas de la nube deben proporcionar ancho de banda a demanda. Además, el crecimiento masivo del ancho de banda requiere una reducción de 10 a 100 veces en el costo y el consumo de energía por bit. Por lo tanto, algunos de los requisitos y desafíos para las redes de comunicación óptica de la próxima generación son: alto ancho de banda, baja latencia, alta confiabilidad, aprovisionamiento rápido de servicios y colaboración abierta [1].

Los consumidores no solo exigen cada vez más medios digitales, sino que también requieren que se entreguen a velocidades cada vez más altas. La demanda exponencialmente creciente de velocidades de transmisión cada vez más altas se ve impulsada por aplicaciones centradas en el vídeo, como por ejemplo: Netflix, YouTube, Facebook e Instagram entre otras. El incremento en el consumo del ancho de banda se ve agravado por el hecho de que en muchos hogares existen múltiples dispositivos (PCs, TVs, tabletas, teléfonos inteligentes, electrodomésticos) conectados a la red, por lo que el consumo se multiplica simultáneamente,

y cada usuario final seguirá exigiendo la misma calidad de servicio. Se prevé que para el 2021 habrá en todo el mundo 3,5 dispositivos conectados en red per cápita.

La evidencia de la creciente demanda de ancho de banda está respaldada por varios estudios; Cisco, el año pasado, pronosticó en su informe de impacto de las aplicaciones en las redes [2] que el tráfico global de Internet se incrementará 127 veces desde 2005 hasta 2021, que el tráfico global promedio de Internet se triplicará de 2016 a 2021, mientras que en el mismo período, el tráfico de hora punta, se multiplicará por 4,6. Para 2021, las conexiones máquina a máquina (p. ej. medidores inteligentes, videovigilancia, monitoreo de atención médica, transporte y seguimiento de paquetes) serán la categoría de más rápido crecimiento, con un aumento de casi 2,4 veces durante el período 2016 a 2021, y serán el 51 % del total de dispositivos y conexiones. A este crecimiento le seguirá el de los teléfonos inteligentes que para 2021 se estima que serán el 23 % del total de dispositivos y conexiones. A pesar de este crecimiento, se estima que sólo el 5 % del tráfico de Internet global para el 2021 se deberá a los dispositivos máquina a máquina, que el 39 % se deberá a los teléfonos inteligentes, el 28 % a los PCs y el 19 % a TVs (que incluyen televisores de panel plano, decodificadores, adaptadores de medios digitales, reproductores de discos Blu-ray y consolas de juegos).

En la Figura 1.1 se muestra un diagrama de conectividad de extremo a extremo, el cual se puede dividir en tres redes principales: la red de acceso (*Access Network* - AN), la de área metropolitana (*Metropolitan-Area Network* - MAN) y la red troncal de área amplia (*Wide-Area Network* - WAN). La red de acceso es responsable de recopilar el tráfico del usuario final y, por lo general, su extensión es de unos pocos kilómetros (en un máximo de 10 a 20 km). Entre los ejemplos de redes de acceso se incluyen redes Ethernet internas y redes operadas por proveedores locales de fibra y cable. La MAN es responsable de transportar el tráfico entre diferentes redes de acceso y de enrutar parte del tráfico a la red troncal WAN. La MAN generalmente se extiende desde unas decenas hasta unos cientos de kilómetros de cobertura. La WAN troncal de larga distancia interconecta a las MAN, y sus enlaces pueden abarcar desde unos pocos cientos hasta miles de kilómetros [3].

Para las redes de acceso óptico (*Optical Access Networks* - OANs) de bajo coste son clave las redes óptica pasiva (*Passive Optical Networks* - PONs) ya que son altamente rentables. En la actualidad se utilizan dos tipos de tecnologías para implementar PON: las basadas en multiplexación por división de tiempo (*Time Division Multiplexing* - TDM) y las basadas en multiplexación por división de longitud de onda (*Wavelength Division Multiplexing* - WDM)

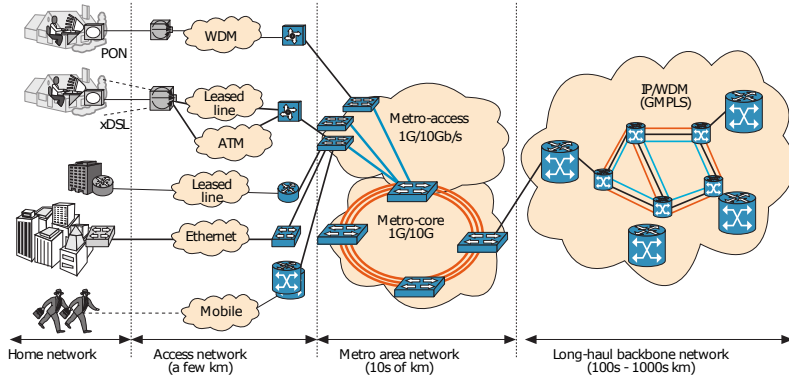


Figura 1.1: Diagrama de conectividad de extremo a extremo: AN, MAN, WAN. Fuente: [3].

[4]. La arquitectura TDM-PON necesita algoritmos de conmutación complejos y tecnología de trama para admitir una variedad de servicios. Por otro lado, WDM-PON asigna datos de alta velocidad a diferentes longitudes de onda, lo que aumenta el coste y la complejidad del sistema [5, 6].

Sin embargo, la reducción de costos ofrecida por una PON podría no ser suficiente para la futura red de telecomunicaciones [7]. Un PON con mayor alcance puede mejorar aún más la rentabilidad de la OAN [8-10]. La PON de largo alcance (*Long-Reach* - LR) extiende el alcance de cobertura de las PON desde el rango tradicional de 20 km hasta 100 km y más, aprovechando los amplificadores ópticos en el medio de los tramos de fibra de transmisión. LR-PON combina AN y MAN en un sistema integrado. La fusión de estos dos tipos de redes simplifica la arquitectura jerárquica de la red de telecomunicaciones, con el acceso de largo alcance vinculado directamente a la red troncal. Esta fusión reduce el número de interfaces de equipos, elementos de red e incluso nodos, lo que resulta en un ahorro de energía [11].

1.1 OFDM para comunicaciones ópticas

La modulación basada en la multiplexación por división de frecuencia ortogonal (*Orthogonal Frequency Division Multiplexing* - OFDM) ha sido adoptada en la mayor parte de los nuevos estándares de comunicaciones alámbricas e inalámbricas de banda ancha ya que brinda una solución simple a la compensación de las distorsiones del canal, logrando al mismo tiempo una alta eficiencia espectral. En la actualidad existen varios estándares de comunicacio-

nes que aplican este tipo de modulación y a continuación se enumeran los más relevantes. LTE (*Long Term Evolution*) [12] utilizado por la cuarta generación de tecnologías de telefonía móvil (4G), la televisión digital terrestre en sus diversas normas DVB-T (*Digital Video Broadcasting-Terrestrial*) [13], DVB-H (*DVB Handheld*) [14], DVB-SH (*DVB Satellite services to Handhelds*) [15] y DVB-T2 (*DVB Second Generation Terrestrial*) [16]; la norma WiMAX (*World-wide Interoperability for Microwaves Access*) [17] utilizada en la transmisión inalámbrica de datos en MAN, el estándar IEEE 802.11 para redes WLAN (*Wireless Local Area Networks*) [18, 19], y los estándares ADSL (*Asymmetric Digital Subscriber Line*) o VDSL2 (*Very high speed Digital Dubscriber Line 2*) [20] para el acceso a internet de banda ancha, entre otros.

Recientemente en las comunicaciones ópticas, dado el continuo incremento de las velocidades de transmisión hasta las decenas de Gb/s, se ha despertado el interés por el uso de OFDM para simplificar la etapa de compensación electrónica de distorsiones (*Electronic Dispersion Compensation - EDC*) [21, 22], permitiendo al mismo tiempo el aumento de la tasa binaria sin necesidad de modificar la estructura de las redes ópticas, actuando únicamente en los equipos transmisor y receptor. Se ha comprobado que la utilización de OFDM permite compensar la dispersión cromática (*Chromatic Dispersion - CD*) y la dispersión en modo de polarización (*Polarization-Mode Dispersion - PMD*) en fibras monomodo y la dispersión modal en fibras multimodo [5, 6]. La ventaja de emplear técnicas EDC en recepción como hace OFDM, frente a las tradicionales de uso de fibra de dispersión cromática o técnicas de precompensación en transmisión, hace que la solución en el receptor sea más flexible, ya que no requiere del conocimiento del canal óptico en el transmisor y facilita el uso de redes conmutadas ópticamente.

Además, en comparación con el OFDM tradicional, el uso de modulación adaptativa (*Adaptively Modulated - AM*) sobre OFDM en sistemas ópticos (*Optical OFDM - OOFDM*) puede mejorar aún más la capacidad de transmisión de la señal y la flexibilidad de la red, sin aumentar el coste del sistema. En AMOOFDM el tipo y orden de modulación utilizado para cada subportadora de datos se ajusta de acuerdo con las características de transmisión del sistema. La eficiencia espectral (*Spectral Efficiency - SE*) se mejora al asignar un orden de modulación más alto a las subportadoras de “alta calidad” mientras que los órdenes de modulación más bajos se asignan a las subportadoras de “baja calidad” [23, 24]. Una de las modulaciones más utilizadas en OOFDM es la modulación de amplitud en cuadratura (*Quadrature Amplitude Modulation - QAM*) por su gran capacidad de carga, llegando hasta constelaciones de 4096 puntos con la tecnología actual.

Los sistemas OOFDM se pueden clasificar en dos grandes grupos según el método de detección que utilicen: los de intensidad modulada y detección directa (*Intensity Modulation and Direct Detection* - IM/DD) y los de detección coherente. El sistema OFDM óptico coherente (*Coherent OOFDM* - CO-OFDM) proporciona una alta SE, una alta resistencia a la CD de la fibra y a la sensibilidad del receptor, pero requiere una implementación compleja y costosa, lo cual lo hace más adecuado para sistemas de transmisión de larga distancia [25]. El sistema IM/DD OOFDM es ventajoso para aplicaciones de corto alcance (Short-Reach - SR) ya que presenta una implementación simple de bajo coste. De hecho, el desafío de las comunicaciones de SR sobre fibra óptica, es desarrollar soluciones rentables basadas en el uso de componentes de bajo coste y alta velocidad.

El uso de OFDM en comunicaciones ópticas aún no está consolidado, por lo tanto, aún existen un gran número de frentes en los que se debe investigar para hacer factible el uso de esta tecnología en el futuro. En la parte óptica hay que diseñar la etapa de modulación/demodulación ópticas, ya que durante décadas la mayoría de los dispositivos han sido diseñados para transmitir pulsos. Junto a dicha etapa se debe investigar los parámetros de OFDM más convenientes a emplear según las características del canal óptico. Además, en la parte electrónica se debe trabajar en las etapas de procesado digital de señal; por un lado, en las técnicas que compensen la parte más débil de OFDM: su sensibilidad a no-linealidades, las cuales son intrínsecas a un enlace de fibra óptica, y por otro, el diseño de arquitecturas y circuitos VLSI (*Very Large Scale Integration*) de altas prestaciones para la implementación, tanto de OFDM como de las técnicas de compensación, a las altas velocidades requeridas por las comunicaciones ópticas.

Los sistemas sobre fibra óptica disponen de una enorme capacidad y para poder aprovecharlos al máximo se requiere la innovación de nuevos esquemas de amplificación óptica, formato de modulación y diseño de fibra [26]. Se pueden utilizar dos formatos de modulación en las comunicaciones ópticas: modulaciones monoportadora o multiportadora. Se ha demostrado que el OFDM óptico tiene varias ventajas en comparación con el formato de modulación de portadora única [27, 28]. Estas ventajas incluyen: estimación y compensación directa del canal en frecuencia, asignación de cantidad de bits y potencia de forma independientemente en cada subportadora, menor complejidad computacional gracias a la eficiencia del algoritmo de la transformada rápida de Fourier (*Fast Fourier Transform* - FFT), posibilidad de adaptarse a canales que varían en el tiempo, eficiencia espectral, modulación de alto orden, flexibilidad y escalabilidad.

1.1.1 OFDM, origen y evolución

El origen de OFDM data de mediados de los años 60, cuando un investigador de los Laboratorios Bell llamado Robert W. Chang publica un artículo sobre la síntesis de señales limitadas en banda para transmisión multicanal [29]. Este trabajo ha sido el pionero en la transmisión de datos utilizando señales ortogonales, donde por primera vez se evitaba la interferencia entre símbolos (*Inter Symbol Interference - ISI*) y la interferencia entre portadoras (*Inter Carrier Interference - ICI*). Pero no es hasta 1970 que Chang logra patentar la idea para los Laboratorios Bell [30]. En la Figura 1.2 se muestra el diagrama de bloques del transmisor y receptor de multiplexación de frecuencia ortogonal de tres canales con filtros de conformación ideado y patentado por Chang.

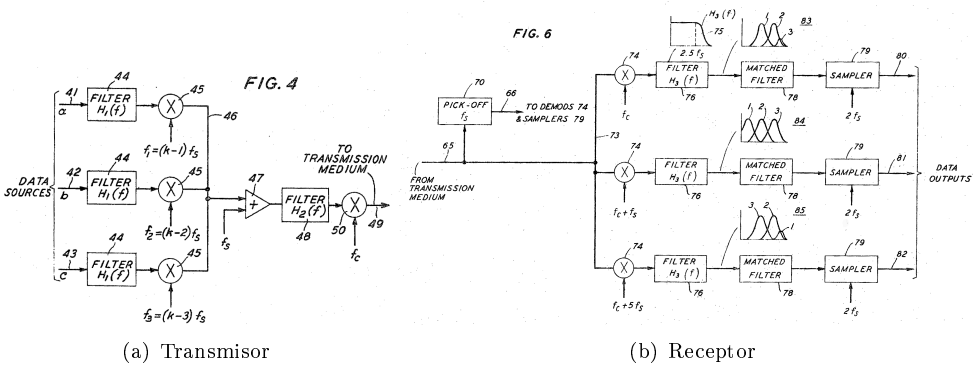


Figura 1.2: Diagrama de bloques del primer sistema de multiplexación de frecuencia ortogonal de tres canales. Fuente: [30].

Para un gran número de canales, los conjuntos de generadores senoidales y demoduladores coherentes requeridos en los sistemas OFDM se vuelven irracionalmente costosos y complejos [31]. En 1971 Weinstein y Ebert, demostraron que una señal de datos multitonal se puede obtener a través de la transformada inversa de Fourier, la cual utiliza una base ortogonal, y que el banco de demoduladores coherentes puede ser reemplazado por la transformada directa de Fourier. Este enfoque ha permitido la implementación de un módem OFDM completamente digital construido alrededor de la FFT y su inversa (*Inverse FFT - IFFT*), eliminando los costosos osciladores y demoduladores, dando origen a los sistemas OFDM modernos de bajo coste.

El tercer aspecto clave de la base de los sistemas OFDM actuales es la incorporación del prefijo cíclico (*Cyclic Prefix - CP*), que fue propuesto por dos

integrantes del departamento de informática de IBM en 1980 [32]. Aunque este prefijo introduce cierta redundancia y, por lo tanto, disminuye la tasa binaria del sistema, se ha demostrado que su uso elimina tanto la ISI como la ICI de la señal recibida y es fundamental para simplificar la tarea de ecualización en OFDM [33].

En [34] se puede ver cómo ha sido la evolución de esta modulación desde sus comienzos hasta llegar a lo que actualmente se denomina OFDM. Desde entonces esta técnica ha fascinado a los investigadores, pero debido a la complejidad de su implementación, su aplicación en estándares de comunicaciones [12-20] tuvo que esperar algunas décadas.

1.1.2 Principios básicos de OOFDM

A continuación se presentan algunos conceptos fundamentales de OFDM de los cuales se hará mención a lo largo de esta tesis doctoral. Esta sección no busca hacer un desarrollo exhaustivo de esta modulación, ya que existe una gran bibliografía acerca de OFDM en comunicaciones ópticas, [26, 33, 35] por citar algunas de las más relevantes. Simplemente se busca destacar algunos aspectos y ecuaciones que serán de utilidad para el desarrollo de este trabajo.

En OFDM el espectro de la señal se divide en varios subcanales y la información modula múltiples subportadoras de diferentes frecuencias, tal y como indica la técnica de multiplexado por división de frecuencia (*Frequency-Division Multiplexing* - FDM), pero de manera tal que las subportadoras sean matemáticamente ortogonales durante todo el periodo del símbolo OFDM [36]. Gracias a esta propiedad de las subportadoras, el canal de banda ancha puede ser interpretado como múltiples canales de banda estrecha simultáneos. El número de subportadoras utilizadas no guarda relación directa con la tasa binaria del sistema, pero sí con la potencia fuera del ancho de banda, lo cual afecta a la interferencia entre bandas adyacentes. Tanto la demodulación, modulación y multiplexación se obtienen digitalmente utilizando la FFT y su inversa. Como resultado, las señales ortogonales se pueden generar y recuperar con precisión y de manera computacionalmente eficiente [33]. En la Figura 1.3 se presenta un diagrama de bloques de un sistema completo OFDM para comunicaciones ópticas.

Los datos a transmitir se ingresan al transmisor en formato binario y de manera serial, a continuación son paralelizados para que pequeños grupos de bits sean asignados, previa transformación, a las subportadoras activas (N_{usc}) que integran un mismo símbolo OFDM. La cantidad de bits que se asignan a cada

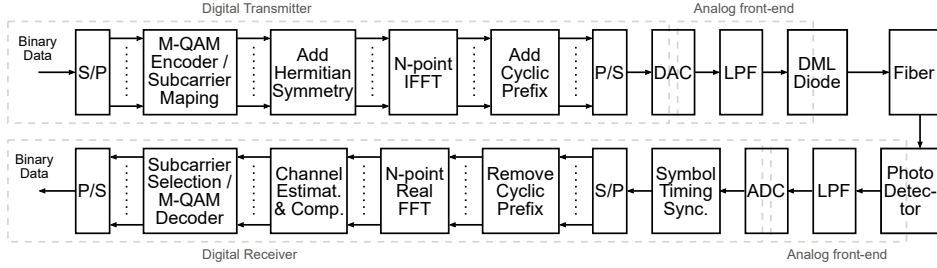


Figura 1.3: Diagrama de bloques de un sistema IM/DD PON OOFDM.

subportadora dependerá del orden (M) y tipo de modulación utilizada (QAM), el cual estará limitado por la relación de señal a ruido (*Signal-to-Noise Ratio* - SNR) del canal. Cada conjunto de bits se transforma en un valor complejo tras su paso por un mapeador, los cuales son utilizados para cargar las subportadoras activas, las subportadoras no activas se cargan con ceros. La señal eléctrica en un sistema IM/DD OOFDM debe ser real, por lo tanto, se debe cumplir la simetría hermitica en el dominio de la frecuencia, lo cual se logra completando las $N/2$ subportadoras de índice negativo con el complejo conjugado de las $N/2$ subportadoras de índice positivo (siendo N el número total de subportadoras). El paso del dominio de la frecuencia al tiempo se realiza mediante la IFFT, que a su vez garantiza la ortogonalidad de las subportadoras. Por último, antes de convertir la señal digital en eléctrica, se escogen un cierto número de muestras (N_{cp}) del final del símbolo OFDM y se lo inserta al comienzo del mismo, a estas muestras se las denomina prefijo cíclico. La longitud de este prefijo depende de la dispersión del canal.

La frecuencia de muestreo (F_s) utilizada por el DAC y el ADC, junto con el número de subportadoras del símbolo OFDM (es decir, el número de puntos de la FFT/IFFT), determinan la separación en frecuencia (F_s/N) de las subportadoras. De manera similar, el número de subportadoras activas determina el ancho de banda (*Bandwidth* - BW) de la señal OFDM, tal como describe la ecuación 1.1. La señal eléctrica a la salida del DAC tiene un ancho de banda mayor al establecido por la ecuación 1.1, esto se debe a las réplicas atenuadas del espectro inherentes al proceso de conversión digital analógico, por lo tanto debe ser limitado utilizando un filtro pasa bajos (*Low Pass Filter* - LPF).

$$BW = N_{usc} \cdot \frac{F_s}{N} \quad (1.1)$$

A continuación la señal eléctrica se convierte en óptica mediante la modulación de la intensidad de una portadora óptica (diodo láser), algo similar a lo que ocurre en radio frecuencia con la modulación de amplitud. Existen tres tipos fundamentales de moduladores de intensidad ópticos: modulador láser directo (*Directly Modulated Laser* - DML), modulador basado en electroabsorción (*Electro-Absorption Modulator* - EAM) y modulador Mach-Zehnder (*Mach-Zehnder Modulator* - MZM). Cada uno de estos moduladores tiene sus ventajas y desventajas, y todos ellos presentan alinealidades que distorsionan la señal OFDM y su efecto se puede modelar como una interferencia de mezclado entre subportadoras (*Subcarrier-to-Subcarrier Intermixing Interference* - SSII). Existen diversas técnicas [37-40] para moderar las alinealidades de cada uno de los moduladores, lo cual trae aparejado un incremento de la complejidad computacional del transmisor y/o receptor OFDM.

En el receptor OFDM de la Figura 1.3 se observa una primera etapa (compuesta por el fotodector, el LPF y el ADC) que es la encargada de la demodulación y conversión de la señal óptica a eléctrica, del filtrado antialias y de la digitalización de la señal OFDM para su posterior tratamiento digital, mediante el cual se podrán obtener los bits de información recibidos. Lo primero y más importante antes de comenzar a procesar la señal, es determinar dónde comienza cada símbolo OFDM, de forma tal que se puede aplicar correctamente la FFT. Para tal tarea se utiliza un sincronizador de trama temporal que se basa en la utilización de un preámbulo conocido. La contaminación inter-símbolo que se produce como consecuencia de la dispersión del canal y el error en la detección del comienzo de trama se ven mitigados por la existencia del CP, el cual es descartado al aplicar la FFT a los símbolos OFDM recibidos.

La estimación y compensación del canal se realiza en el dominio de la frecuencia de manera muy sencilla, siendo ésta una de las ventajas de utilizar OFDM. Para este fin, se transmiten en el preámbulo unos símbolos OFDM conocidos y a partir de lo obtenido en el receptor, se realiza la estimación del canal. Aprovechando la existencia de un bloque de FFT en el receptor, la estimación y compensación del canal se suelen hacer en el dominio de la frecuencia. La validez de la estimación depende de la robustez del algoritmo empleado y de la invariabilidad del canal (p. ej. en los cables de fibra óptica aéreos se producen dispersión y múltiples cambios de atenuación debido al viento), lo cual determina cada cuantos símbolos OFDM se debe volver a estimar el canal y, por lo tanto, afecta ligeramente a la tasa binaria del sistema.

Por último, en el receptor de la Figura 1.3 se encuentra el bloque de desmapeo, el cual convierte los valores complejos de cada subportadora en valores binarios. A continuación podrían incluirse otros bloques, como por ejemplo un corrector

de errores para llevar la tasa de errores a valores muy bajos que permitan considerar la transmisión libre de errores, o un bloque que monitorice la calidad del sistema de comunicación.

La magnitud del vector de error (*Error Vector Magnitude* - EVM) es una medida de calidad de la modulación y se puede calcular mediante la ecuación 1.2, donde $|E_t|$ es la potencia de la constelación ideal normalizada o constelación transmitida, $E_{r,i}$ es el vector de señal la recibida, $E_{t,i}$ es el vector de señal transmitida y N_s es el número de símbolos transmitidos [41]. El EVM nos permite evaluar la calidad de un sistema de comunicaciones a través de la medición de la dispersión de los símbolos recibidos. En un sistema OFDM, se mide el EVM promedio por subportadora y de esta manera se determina la calidad de la modulación para las diferentes frecuencias.

$$\text{EVM} = \frac{\sigma_{err}}{|E_t|}, \quad \sigma_{err}^2 = \frac{1}{N_s} \sum_{i=1}^{N_s} |E_{err,i}|^2, \quad E_{err,i} = E_{r,i} - E_{t,i} \quad (1.2)$$

Una de las grandes desventajas de los sistemas OFDM es que la señal transmitida presenta una gran relación entre la potencia de pico y la potencia media (*Peak-to-Average Power Ratio* - PAPR). Existen diferentes técnicas para abordar la reducción de la PAPR en sistemas OFDM [42] y es un tema que aún continúa en investigación [43]. No existe una solución universal de reducción de PAPR, más bien, la técnica de reducción de PAPR debe ser cuidadosamente elegida de acuerdo a las características del sistema. Tal vez una de las más difundidas y menos costosas desde el punto de vista computacional sea la de recorte o *clipping*. Técnicas más sofisticadas como *DFT-Spread* [44] aplican la transformada discreta de Fourier (*Discrete Fourier Transform* - DFT) a las subportadoras antes de ingresar al módulo IFFT del transmisor y realizan la tarea inversa en el receptor, aumentando de forma significativa el coste computacional del sistema.

Las prestaciones de los sistemas de comunicaciones se suelen evaluar en torno a dos aspectos, uno es la tasa binaria o *bitrate* y el otro es la eficiencia espectral. Este trabajo de tesis tiene por objetivo maximizar estos dos parámetros a través de un estudio minucioso de todos los componentes y algoritmos que los afectan. A continuación se presentan las ecuaciones utilizadas para realizar dichas mediciones y se comentan brevemente algunas particularidades. La tasa binaria se define, en su versión más simple o pura, como el número de bits transmitidos en un símbolo OFDM partido el tiempo de duración del mismo; tal y como se describe en la ecuación 1.3. Si se analiza dicha ecuación se puede

concluir que la tasa binaria en un sistema OFDM depende de 4 factores: M , F_s , N_{usc}/N y N/N_{cp} ; donde el orden de modulación y la tasa de muestreo tienen mayor peso. La longitud del CP y el número de subportadoras activas dependen del canal.

$$\text{bitrate} = \frac{\text{núm. de bits transmitidos}}{\text{tiempo del símbolo}} = \log_2(M) \cdot F_s \cdot \frac{N_{usc}}{N_{cp} + N} \quad (1.3)$$

Esta definición se puede refinar incorporando la pérdida de velocidad debido a la utilización de un preámbulo (utilizado para la sincronización temporal y la estimación de canal), y sumado a esto, se puede contemplar la pérdida de velocidad debido a la redundancia introducida por el corrector de errores. La ecuación 1.3 es válida para el caso en que todas las subportadoras utilizadas empleen la misma carga binaria y como se ha mencionado anteriormente, una de las maneras de aumentar los valores de N_{usc} y M es utilizar la carga adaptativa. Por lo tanto, teniendo en cuenta estos tres aspectos, la tasa binaria se puede redefinir mediante la ecuación 1.4, donde N_s es el número de símbolos transmitidos, N_{ts} es el tamaño del preámbulo y R_b es el ratio del código corrector de errores.

$$\text{bitrate} = \left[\sum_{k=1}^{N_{usc}} \log_2(M_k) \cdot \frac{F_s}{N_{cp} + N} \right] \cdot \left[\frac{N_s \cdot (N_{cp} + N)}{N_{ts} + N_s \cdot (N_{cp} + N)} \right] \cdot R_b \quad (1.4)$$

Como se ha mencionado, la eficiencia espectral es la otra variable que permite medir las prestaciones de un sistema de comunicaciones y se define como el cociente entre la tasa binaria y el ancho de banda de la señal. En la ecuación 1.5 se define la SE para un sistema OFDM y de ella se desprende que esta variable depende principalmente de M y, en menor medida, del cociente N/N_{cp} . En sistemas IM/DD OOFDM que utilizan fibra monomodo estándar (*Standard Single-Mode Fiber* - SSMF) de corta y mediana longitud, la dispersión del canal es muy baja y, por lo tanto, se obtienen valores muy pequeños de N_{cp} en relación con N . Por este motivo y en estas circunstancias, se puede afirmar que la SE depende casi exclusivamente del orden de la modulación utilizada.

$$\text{SE} = \frac{\text{bitrate}}{\text{BW}} = \log_2(M) \cdot \frac{N}{(N_{cp} + N)} \quad (1.5)$$

1.2 Antecedentes y estado actual

Tal y como se ha comentado anteriormente, en las últimas dos décadas se ha despertado el interés por el uso de OFDM en sistemas de comunicaciones ópticas para simplificar la etapa de compensación de EDC, PMD y CD. Sin embargo, aún existen un gran número de frentes en los que se debe investigar para hacer factible el uso de esta tecnología en el futuro.

En la parte óptica, hay que diseñar la etapa de modulación-demodulación ópticas; la solución CO-OFDM es la más inmediata, es equivalente a la solución empleada en comunicaciones inalámbricas donde la portadora de radio frecuencia es sustituida por una portadora óptica [45]. Su puesta en práctica conlleva numerosos problemas debido al uso de un receptor coherente y a la sensibilidad de OFDM a desviaciones en frecuencia de la fuente láser y al ruido de fase de la misma, lo que hace necesario la utilización de algoritmos de seguimiento de frecuencia y fase en el receptor complicando su implementación práctica. La alternativa a CO-OFDM es un campo abierto en el que se han planteado diversas soluciones adaptadas a las características de los sistemas ópticos, cada una con sus ventajas e inconvenientes, siendo en general propuestas que no aprovechan la potencial alta SE de la modulación OFDM. Tal vez las más abordadas sean las soluciones basadas en IM/DD OOFDM que representan la señal OFDM con valores reales (simetría Hermitiana) y solo aprovechan la mitad del ancho de banda. En los sistemas IM/DD, la señal OFDM eléctrica (real y bipolar) modula la intensidad de luz del láser.

Como la intensidad de luz no puede ser negativa, es necesario convertir la señal eléctrica en real y positiva. Existen diferentes técnicas [46] para realizar esto, están las que incorporan un nivel de continua (*DC biased Optical OFDM* - DCO-OFDM), las que sólo transmiten la parte positiva de la señal OFDM (*Asymmetrically Clipped Optical OFDM* - ACO-OFDM) o las que combinan estas dos técnicas (*Asymmetrically clipped DC biased optical OFDM* - ADO-OFDM). Algunas de estas técnicas [47-49] permiten disminuir la potencia media óptica transmitida, disminuir la distorsión armónica producto de las alinealidades del modulador láser y disminuir el ruido debido al recorte de amplitud, entre otros. A cambio disminuyen la eficiencia espectral (p. ej. para ACO-OFDM la SE es la mitad) y aumentan la complejidad del transmisor/receptor OFDM (p. ej. para ADO-OFDM se requiere un bloque IFFT adicional en el transmisor y 3 bloques FFT/IFFT adicionales en el receptor) [46]. Estos bloques adicionales dan como resultado una excesiva complejidad al sistema, lo que aumenta considerablemente el consumo de energía y el área del dispositivo utilizada. Para mejorar las eficiencias de potencia óptica y es-

pectral con una complejidad moderada, un enfoque es aprovechar la SE de DCO-OFDM y desarrollar un procesamiento de señal adecuado para superar la limitación de eficiencia de potencia óptica causada por el ruido de recorte y la excesiva polarización de corriente continua presente en los sistemas DCO-OFDM tradicionales [50]. Este es un campo que continúa en investigación, como lo muestran algunas publicaciones recientes [49, 51, 52], y muchos de sus resultados están siendo aplicados a sistemas de comunicaciones inalámbricas.

En sistemas IM/DD OOFDM de tiempo real (*Real-Time* - RT), que no utilizan WDM o TDM, el incremento de la tasa binaria se puede alcanzar mediante dos enfoques: mejorando la SE y/o aumentando el ancho de banda de señal. El uso de un BW más amplio requiere de conversores digital analógico (*Digital-to-Analog Converters* - DACs) y conversores analógico digital (*Analog-to-Digital Converters* - ADCs) con tasas de muestreo más altas, lo cual implica conversores más caros y con mayores alinealidades, y al mismo tiempo eleva la complejidad de la implementación del sistema sobre dispositivos de lógica programable (*Field Programmable Gate Array* - FPGA) debido a su elevada frecuencia de trabajo. A su vez, este aumento del BW incrementa el coste de los dispositivos analógicos y ópticos necesarios para el tratamiento de la señal electro-óptica. Por otro lado, mejorar la SE requiere el uso de formatos de modulación de alto nivel. Para maximizar el rendimiento del sistema, diferentes subportadoras dentro de un símbolo OFDM pueden cargarse de manera adaptativa con diferentes formatos de modulación de señal de acuerdo con la SNR de cada subportadora [53]. Sin embargo, los formatos de modulación de alto nivel son muy sensibles al ruido de cuantización de los DAC y ADC, que a su vez depende del número efectivo de bits (*Effective Number Of Bits* - ENOB). En los conversores digitales comerciales de alta velocidad, este valor es más bajo que el número de bits nominal, y disminuye conforme aumenta la frecuencia de señal y/o la frecuencia de muestreo. A veces, la resolución de los conversores digitales se mejora utilizando sobremuestreo, es decir, una frecuencia de muestreo varias veces mayor que la tasa de Nyquist, a costa de no utilizar todo el BW disponible de los conversores digitales.

En los últimos años se han realizado muchos esfuerzos por mejorar las prestaciones de los sistemas IM/DD OOFDM en redes PON para cortas y medianas distancias. En [54-56] se presentan sistemas de IM/DD OOFDM que han empleado formatos de modulación 64-, 256-, 1024-QAM, en los que la señal eléctrica recibida fue capturada por un Osciloscopio de muestreo digital (*Digital Sampling Oscilloscope* - DSO) y, posteriormente, las muestras han sido procesadas en un ordenador. Por otra parte, una implementación de hardware en tiempo real debe tener en cuenta requisitos adicionales como la precisión del

punto fijo, los recursos *hardware* disponibles o la velocidad del reloj. Recientemente, algunos sistemas IM/DD OOFDM en tiempo real han sido demostrados experimentalmente [4, 57-62]. Algunos de estos sistemas [54-56, 58, 63] hacen uso de un factor de sobremuestreo entre 2 y 5 para reducir el ruido de cuantificación, mejorar la respuesta de frecuencia analógica de los conversores digitales y relajar las restricciones de diseño del filtro antialias. Sin embargo, esta solución implica un exceso de costos en los conversores digitales y una infrautilización del BW disponible. Para mejorar la SE, esos trabajos emplean *M*-QAM con altos niveles de modulación.

Tras el estudio del estado del arte, se puede afirmar que el sistema IM/DD OOFDM en tiempo real de 10.44 Gb/s [60] que utiliza modulaciones de 16 a 256-QAM sobre 20 km de SSMF, es la tasa binaria más alta que se ha demostrado para un sistema en tiempo real IM/DD OOFDM PON (de una sola banda y una sola longitud de onda) de bajo coste. Hay otros sistemas con velocidades de bits más altas, donde el procesamiento digital de la señal no se realiza en tiempo real [54-56, 64], o se usan componentes muy costosos [65], o técnicas como FDM [66, 67] o WDM [68-70].

1.3 Motivación y objetivos

Aunque OFDM ha sido implementado con éxito en aplicaciones inalámbricas, con tasas de Mb/s, el paso a comunicaciones ópticas con tasas de Gb/s no es inmediato. Con la aparición de conversores de alta velocidad y la gran capacidad de computación de las FPGA actuales, es posible plantearse la implementación en tiempo real de un transmisor/receptor OFDM para comunicaciones ópticas con gran ancho de banda.

El objetivo principal de esta tesis doctoral es maximizar la tasa binaria y la eficiencia espectral de un sistema IM/DD OOFDM de bajo coste, aprovechando al máximo el ancho de banda disponible (límite de Nyquist) y utilizando el mayor orden de modulación posible en cada subportadora. Para tal fin es necesario poder seleccionar y caracterizar cada uno de los dispositivos (ADC, DAC, DML, amplificador, LPF, fibra) que integran en el sistema de comunicación y buscar los algoritmos de DSP adecuados que permitan aumentar las prestaciones y que sean factibles de implementar en un sistema de tiempo real.

Para llevar adelante estas tareas se han fijado los siguientes objetivos:

- Elaboración de un banco de pruebas para caracterizar cada uno de los dispositivos que integran en el sistema de comunicación. El mismo debe

permitir generar y capturar señales de gran ancho de banda a una tasa de 5 GS/s.

- Búsqueda, selección y caracterización de los diferentes dispositivos (*front-end* analógico, conversión electro-óptica y fibra).
- Investigación sobre algoritmos de DSP para sistemas OOFDM, cuya implementación pueda ser paralelizable y cuya complejidad *hardware* no sea excesivamente elevada.
- Elaborar un simulador del sistema completo (algoritmos + canal) en MATLAB que permita probar diferentes técnicas y ver su impacto en las prestaciones del sistema.
- Descripción *hardware* del transmisor y receptor OFDM, utilizando recursos dedicados de la FPGA para aumentar la frecuencia de trabajo y análisis de los recursos utilizados.
- Desarrollar un banco de pruebas óptico para estudiar la viabilidad y las prestaciones de un sistema IM/DD OOFDM de bajo coste y en tiempo real controlado desde MATLAB.
- Optimizar las prestaciones del módem OFDM analizando el EVM por subportadora, evaluando la respuesta del canal, modificando la carga de las subportadoras, ajustando el factor de recorte y las amplitudes de las señales eléctricas.

1.4 Metodología de trabajo

En esta tesis se ha trabajado en las etapas de procesado digital de señal necesarias para poder llevar a cabo una transmisión y recepción en tiempo real. Para ello primero ha sido necesario seleccionar algoritmos de baja complejidad, pero suficientemente robustos, teniendo en cuenta las limitaciones de los componentes (analógicas, ópticas y digitales), para posteriormente poder abordar el diseño de arquitecturas de altas prestaciones que permitan su implementación sobre FPGA. En el diseño de circuitos de alta velocidad son clave la paralelización y la reducción del número de bits empleados, por lo tanto, ha sido necesario hacer un estudio detallado de los bits necesarios en cada etapa (p. ej. diseñando preámbulos que permitan reducir el número de bits en la etapa de sincronización), así como en qué medida es posible paralelizar cada etapa. Para ello se han seguido los siguientes pasos:

- Modelado de algoritmos en MATLAB para el análisis de prestaciones del sistema OOFDM y validación las propuestas algorítmicas.
- Análisis de precisión finita de los algoritmos y su impacto en las prestaciones del sistema OOFDM.
- Estimación de recursos *hardware* de cada uno de los bloques para facilitar las comparaciones con otras arquitecturas y poder tomar decisiones en etapas tempranas del diseño.
- Transformaciones de arquitecturas secuenciales a paralelas, mejorando la velocidad del diseño mediante segmentación y control de *fanout*.
- Descripción *hardware* utilizando Verilog de los diferentes bloques que integran el transmisor y receptor OOFDM. Síntesis, implementación y verificación de las prestaciones física: área y velocidad.
- Validación del sistema mediante medidas realizadas en un banco de pruebas de comunicaciones ópticas.

1.5 Contribuciones de la tesis

La principal contribución de esta tesis es el desarrollado de la arquitectura *hardware* de un receptor OFDM de tiempo real y alta velocidad que se ha utilizado en un banco de pruebas experimental para demostrar la transmisión de símbolos OFDM, cuyas subportadoras han sido moduladas con M -QAM (con M desde 16 hasta 512), utilizando un DML de bajo coste y detección directa sobre 10 km SSMF sin amplificador óptico. En esta configuración, se ha obtenido una SE de hasta 8,38 bit/s/Hz y una tasa binaria de hasta 20,37 Gb/s, convirtiéndolo en el sistema IM/DD OOFDM de bajo coste y en tiempo real con mejores prestaciones hasta la fecha. Tanto el DAC como el ADC trabajan a su máxima tasa de muestreo (5 GS/s) y el BW de la señal OFDM es prácticamente el máximo según el criterio de Nyquist (aproximadamente 2.5 GHz).

A continuación se enumeran las principales aportaciones de esta tesis:

1. Un algoritmo de sincronización temporal basado en la correlación cruzada entre la señal recibida y un preámbulo conocido, que reduce el número de *hardware* aritméticos utilizados y que alcanza excelentes prestaciones en escenarios de comunicaciones ópticas de baja SNR.

2. Un procesador de FFT de longitud variable, con entradas y salidas en orden natural, con una arquitectura paralela segmentada que permite alcanzar hasta una tasa de 10 GS/s en dispositivos Virtex-7 con una eficiencia (área/velocidad) superior a la de otros trabajos publicados. Una primer versión de este procesador se ha desarrollado durante la realización del trabajo de fin de máster.
3. Un método de estimación y compensación de canal en frecuencia, que utilizada diezmado para disminuir el tamaño del preámbulo y la complejidad *hardware*, y que utiliza repetición para aumentar la precisión de la estimación.
4. La viabilidad de un sistema de bajo coste en tiempo real que mediante el aprovechamiento del ancho de banda disponible y de la carga adaptativa de las subportadoras ha permitido casi duplicar la eficiencia espectral y la tasa binaria del sistema con mejores prestaciones publicado hasta la fecha.

De manera colateral, se ha obtenido un generador y adquisidor de señales arbitrarias con una tasa de muestreo máxima de 5 GS/s que se conecta a un ordenador mediante un interfaz gigabit ethernet, el cual permite probar y experimentar, con otro tipo de modulaciones en diferentes entornos ópticos.

Además, durante los años de investigación de esta tesis, el trabajo realizado ha derivado en diversas publicaciones en revistas y congresos, que a continuación se resumen:

- Revistas del primer cuartil
 - Julián S. Bruno, Vicenç Almenar, Javier Valls, y Juan L. Corral, “Real-time 20.37 Gb/s optical OFDM receiver for PON IM/DD systems” *Opt. Express* 26, 18817-18831 (2018).
 - J. S. Bruno, V. Almenar, J. Valls y J. L. Corral, “Low-complexity time synchronization algorithm for optical OFDM PON system using a directly modulated DFB laser” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, no. 11, pp. 1025-1033, 1 November 2015.
- Otras revistas
 - Julián S. Bruno, Vicenç Almenar y Javier Valls, “FPGA implementation of a 10 GS/s variable-length FFT for OFDM-based optical

communication systems” *Microprocessors and Microsystems*, vol. 64, pp. 195-204, February 2019.

- Congresos internacionales
 - M. Morant, J. S. Bruno, V. Almenar and R. Llorente, “Performance evaluation of OFDM and SC-QAM backhaul provision on FTTH optical access networks including multi-core fiber riser” *Proc. SPIE 10945, Broadband Access Communication Technologies XIII*, 1094508 (4 February 2019).
 - R. Llorente, M. Morant, J. S. Bruno, V. Almenar, J. L. Corral, J. M. Fuster, y J. Valls, “Multidimensional Multiplexing in Multicore Fibre for Hybrid Optical Backhaul provision: The XCORE approach” *20th International Conference on Transparent Optical Networks (ICTON 2018)*.
- Congresos nacionales
 - Julián S. Bruno, Vicenç Almenar, Javier Valls y Juan L. Corral, “Algoritmo paralelo de sincronización de tiempo para sistemas OFDM ópticos utilizando un dispositivo FPGA Virtex-7” *XV Jornadas de Computación Reconfigurable y Aplicaciones (JCRA 2015)*.
 - Julián Santiago Bruno, Javier Valls Coquillat y Vicenç Almenar Terré, “Diseño e Implementación de una FFT de alta velocidad para OFDM en comunicaciones ópticas” *XXIX Simposium Nacional de la Unión Científica Internacional de Radio (URSI 2014)*.

1.6 Organización de la tesis

En esta sección se expone la organización de esta tesis doctoral. En el capítulo 2 se presentan los detalles y la caracterización de la plataforma diseñada para realizar los experimentos necesarios para el desarrollo de la tesis. Además se busca caracterizar cada componente, para conocer sus limitaciones y cómo éstas pueden afectar al sistema OFDM.

El estudio y diseño de los diferentes bloques que integran el transmisor y receptor OOFDM se realiza en el capítulo 3. Allí se describen cada uno de los algoritmos, se explica su modelo de precisión finita y se realizan las simulaciones pertinentes. En ese mismo capítulo se describe un simulador funcional (secuen-

cial y en coma flotante) y un simulador *hardware* (codificación con funciones básicas y en punto fijo) del transmisor y receptor, desarrollado en MATLAB para el cual también se ha modelado el comportamiento de los conversores digitales y la etapa óptica.

El capítulo 4 trata sobre las decisiones de implementación y los recursos de la FPGA que se han utilizado para llevar a cabo, tanto el transmisor como el receptor OFDM, como así también los bancos de pruebas utilizados para verificar el correcto funcionamiento de todo el sistema.

En el capítulo 5 se muestra la plataforma de experimentación, la estimación de canal obtenida para las diferentes configuraciones y los resultados de velocidad y eficiencia espectral para carga de subportadoras constante y variable. También se presentan las características y prestaciones de otros sistemas que se pueden encontrar en la bibliografía de estos últimos años y se realiza un análisis comparativo con el sistema desarrollado en esta tesis.

Finalmente, en el capítulo 6 se presentan las conclusiones de este trabajo de tesis y se exponen las futuras líneas de investigación.

Capítulo 2

Caracterización de la plataforma de experimentación

En este capítulo se presentan los detalles y caracterización de la plataforma diseñada para realizar los experimentos, que se describen más adelante en el capítulo 5, necesarios para el desarrollo de la tesis. Lo que aquí se busca es conocer en profundidad lo que puede dar de sí cada componente y estimar cómo esto puede afectar al sistema de forma tal que a la hora del diseño se tenga en cuenta.

Una de las primeras y principales tareas ha sido la búsqueda, elección, compra y caracterización de todos los dispositivos que integran el sistema de comunicaciones ópticas que tiene como objetivo esta tesis. Si bien es cierto que todos los dispositivos tienen sus propias hojas de características, muchas de ellas con grandes carencias, el principal problema viene dado por el hecho de que no se cuenta con la información suficiente para el rango de frecuencias en el cual se desarrolla el presente trabajo. Un buen ejemplo de esto último es el caso de los convertidores utilizados, la mayoría de las características se dan para frecuencias de muestreo bastante inferiores y no se dispone de una información detallada de la respuesta en frecuencia ni del ENOB, lo cual es primordial para poder obtener las mejores prestaciones del sistema OFDM. Por ello, es necesario caracterizar estos dispositivos en las condiciones de operación del diseño.

Algunos autores, para mejorar las prestaciones de este tipo de sistemas, utilizan dispositivos con un ancho de banda analógico y una frecuencia máxima de muestreo varias veces superior a los del sistema transmisión, de forma tal que muchos efectos no deseados, que aparecen al intentar aprovechar al máximo sus prestaciones (p. ej. ancho de banda), disminuyen drásticamente. Sin embargo, este tipo de decisiones encarece de manera muy significativa el coste de los dispositivos. En el caso que nos ocupa, uno de los objetivos de esta tesis es desarrollar un sistema de bajo coste, razón por la cual, la elección de cada dispositivo ha sido muy cuidadosa y se ha llevado al límite el funcionamiento de cada uno, ayudado por el procesamiento digital de señales incorporado dentro de las FPGAs, con el fin de obtener mejores prestaciones por un coste inferior.

Para poder desarrollar los algoritmos del sistema de transmisión es necesario primero conocer las características de todos los dispositivos que intervienen en el sistema de transmisión, para ello se ha desarrollado un banco de pruebas como el que se muestra en la Figura 2.1. Como se puede observar en dicha figura, se ha desarrollado en las FPGAs un sistema que permite enviar muestras al DAC y recibir muestras del ADC, y todo esto controlado desde el ordenador con una aplicación desarrollada en MATLAB que se comunica con las tarjetas mediante un interfaz de gigabit ethernet.

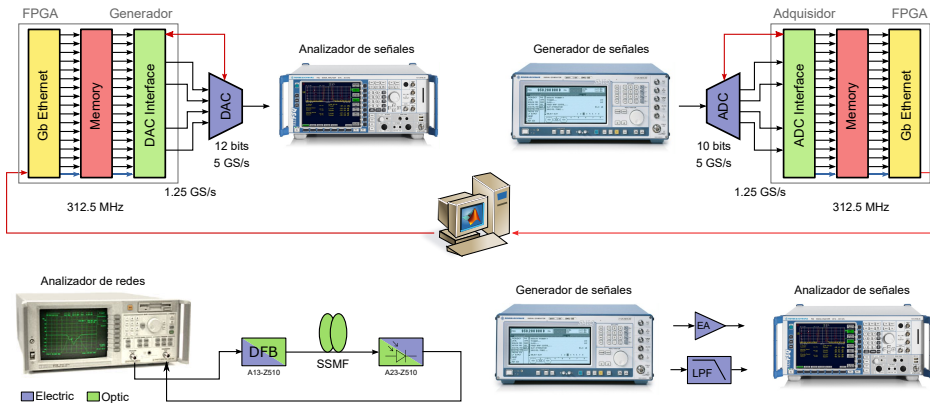


Figura 2.1: Banco de pruebas para caracterización del sistema IM/DD OFDM para comunicación óptica.

A continuación se presentan los detalles de la caracterización de los componentes elegidos para abordar el diseño de la plataforma experimental: el convertor DA, el amplificador eléctrico, la etapa electro-óptica, el filtro antisolapamiento, el convertor AD y el reloj de muestreo.

2.1 Conversor DA

El conversor DA que se va a utilizar en esta plataforma es el MD657B, fabricado por la compañía EUVIs. Es un conversor de 12 bits que puede alcanzar una tasa de muestreo de hasta 5,5 GS/s y posee un rango dinámico libre de espurios (SFDR) típico de entre 45 y 60 dBc, y un error por alinealidades (INL) de ± 2 LSB. La salida analógica es diferencial y puede trabajar en dos modos: *Normal-Hold* o *Return-to-Zero*.

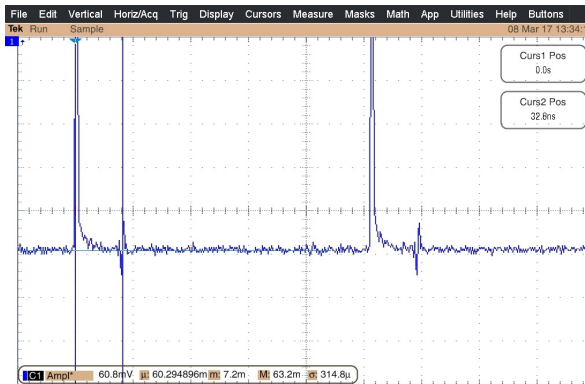
Al comenzar a evaluar las prestaciones del DAC, se ha observado que existe un rizado en la respuesta en frecuencia y que el mismo varía en función de la longitud de los cables conectados a la salida del dispositivo, lo cual sugiere algún tipo de desadaptación de impedancias. Para probar la existencia de una onda reflejada, producto de la desadaptación, se ha generado un pulso rectangular de 200 ns con un ciclo de actividad muy bajo y se ha conectado la salida del DAC a un osciloscopio digital mediante un cable cuya impedancia característica es 50Ω , la misma que el conversor y el osciloscopio. Se realizaron dos mediciones, una utilizando un cable corto y otra con un cable largo, las señales capturadas se muestran en la Figura 2.2. En ambos casos se observa un pequeño impulso con un retardo, en referencia al pulso rectangular generado por el DAC, proporcional a la longitud del cable.

Después de varias pruebas y de un extenso intercambio de ideas con el fabricante, se ha concluido que la única solución para disminuir los efectos de la desadaptación de impedancias es utilizar un atenuador de al menos 10 dB entre la salida del DAC y la entrada de un amplificador eléctrico, de ahora en adelante se denominará a esto salida corregida. La incorporación de este atenuador produce que la onda incidente se atenúe 10 dB y la onda reflejada se atenúe 20 dB, de manera tal que en términos relativos la amplitud de la onda reflejada se vea atenuada 10 dB. El efecto de atenuar la onda reflejada se puede ver manifiesto en la respuesta en frecuencia de la Figura 2.3.b, donde se observa la extinción del rizado presente en la Figura 2.3.a. Todo este proceso ha supuesto varias semanas de trabajo extra que no se pueden ver reflejadas en el resultado final, pero que son propias de una tesis que tiene por objetivo el diseño y la implementación de un sistema en tiempo real.

Un conversor DA ideal tiene una respuesta en frecuencia que se representa mediante la función $\sin(x)/x$, cuyos cruces por cero coinciden con los múltiplos enteros de la frecuencia de muestreo. Esto provoca que el módulo de la respuesta para una frecuencia de $Fs/2$ sea 4 dB menor que para 0 Hz. En la Figura 2.3.a se muestra el módulo de la respuesta en frecuencia del DAC MD657B operando a una tasa de muestreo de 5 GS/s, la cual se obtiene generando una



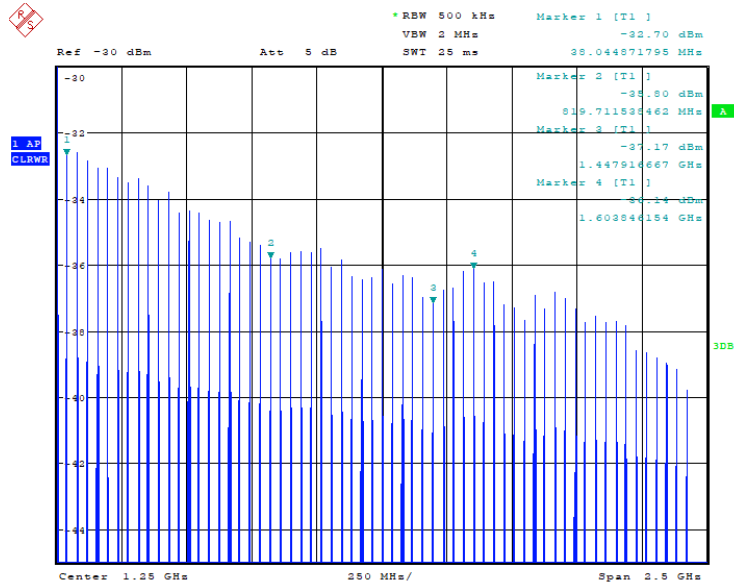
(a) Cable corto



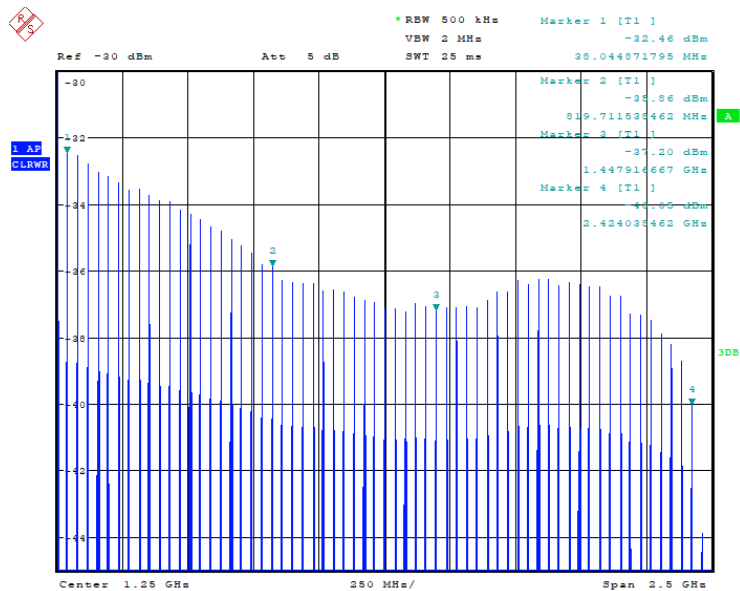
(b) Cable largo

Figura 2.2: Medición de onda reflejada para un pulso rectangular generado con el DAC.

señal multitonal con el convertor y siendo capturada con un analizador de señal (R&S FSQ26). A pesar de la existencia del rizado, se observa que la caída de la respuesta en frecuencia se asemeja a la ideal, pero con mayor pendiente (aproximadamente 7 dB en el extremo de banda). En la Figura 2.3.b se observa que la respuesta en frecuencia para el DAC con la salida corregida se aparta un poco más de la ideal y esto es debido a la respuesta en frecuencia del amplificador utilizado. En este último caso, las altas frecuencias sufren una atenuación de 8 dB con respecto a las bajas frecuencias, lo cual es el doble de lo que sucede en un DAC ideal. Esto no es un asunto menor si se pretende utilizar preénfasis, tema que se debate en la apartado 3.1.3, ya que se deberán atenuar 8 dB las más bajas frecuencias en lugar de 4 dB (caso ideal).



(a) Salida desadaptada



(b) Salida corregida

Figura 2.3: Respuesta en frecuencia del DAC MD657B operando a 5 GS/s.

Otra característica de los convertidores DA es la generación de armónicos y espurios, que son dependientes del rango de frecuencia de la señal generada y de la frecuencia de muestreo utilizada. A continuación se presentan 6 espectros en la Figura 2.4, correspondientes a señales senoidales de diferentes frecuencias, obtenidos con un analizador de señales conectado a la salida del DAC (compensada con amplificador y atenuadores) cuya tasa de muestreo es de 5 GS/s. Si se analizan las 6 gráficas, se observa que la potencia de los armónicos y espurios conserva un nivel inferior a los -40 dB y que la potencia de la señal va disminuyendo desde 0 dB hasta casi -10 dB.

Todo esto nos habla de que los bits efectivos del convertidor son bastante menos que los indicados por la resolución (12 bits), que estos varían en función de la frecuencia de la señal generada y que la atenuación en el extremo de la banda es mucho mayor que los 4 dB esperados. Esta degradación de la SNR del convertidor con el aumento de la frecuencia de la señal afectará, como se verá en el capítulo 5, a la carga binaria que se puede asignar a cada portadora de un símbolo OFDM.

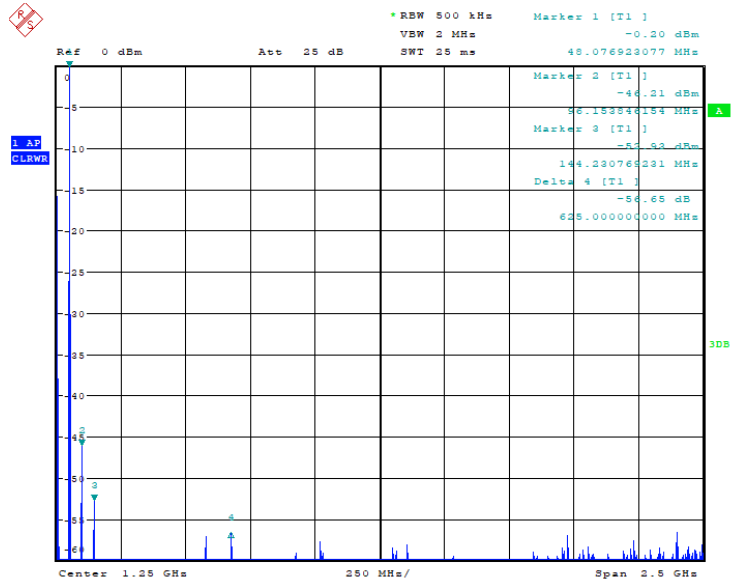
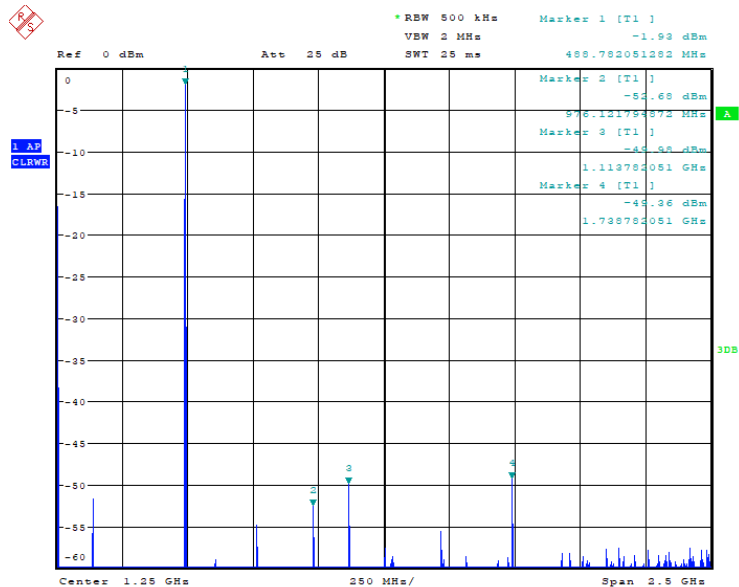
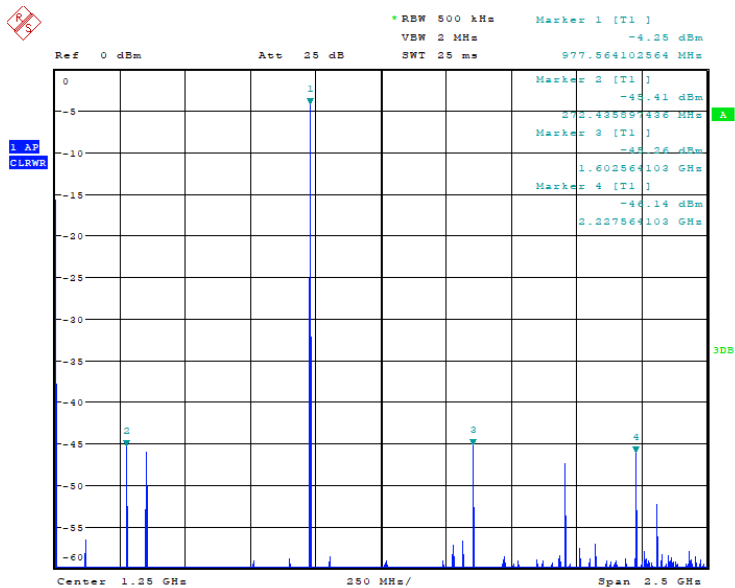
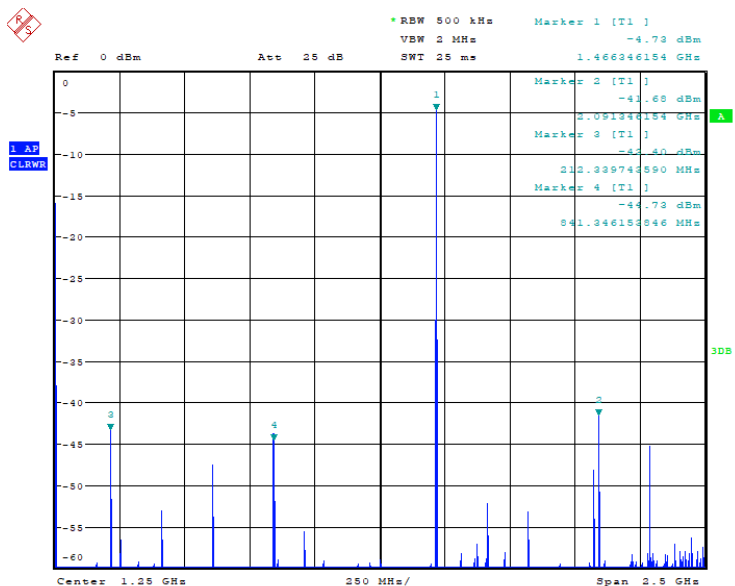
(a) 49 MHz ($10 \cdot 5e9/1024$)(b) 488 MHz ($100 \cdot 5e9/1024$)

Figura 2.4: Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.



(c) 976 MHz ($200 \cdot 5e9 / 1024$)



(d) 1465 MHz ($300 \cdot 5e9 / 1024$)

Figura 2.4: (Continuación) Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.

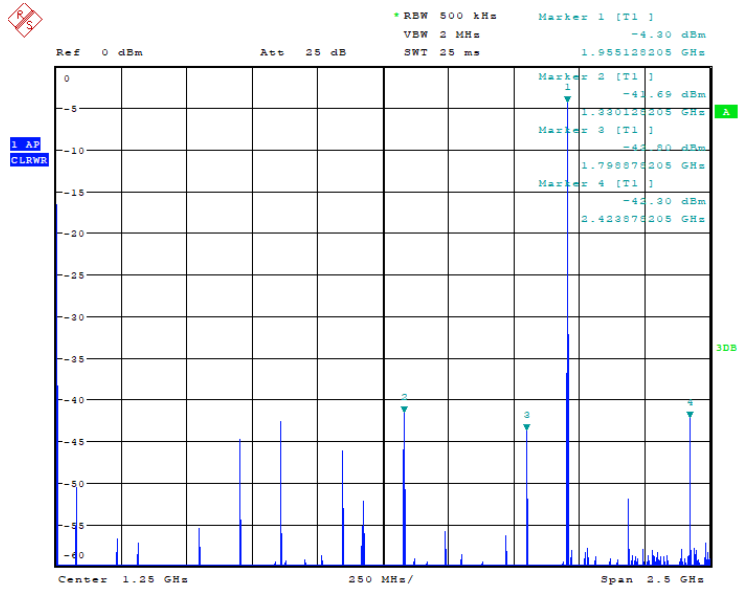
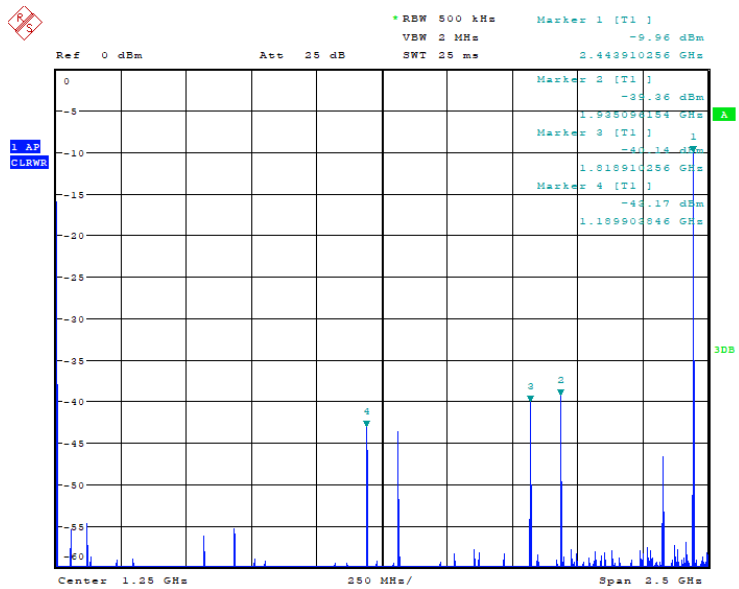
(c) 1953 MHz ($400 \cdot 5e9/1024$)(d) 2441 MHz ($500 \cdot 5e9/1024$)

Figura 2.4: (Continuación) Espectros de la generación de tonos con el DAC MD657B operando a 5 GS/s.

2.2 Amplificador eléctrico

El amplificador eléctrico utilizado en este sistema es el ABP0300-01-2730 del fabricante Wenteq, posee una ganancia típica de 27 dB, una variación de la ganancia de $\pm 0,75$ dB, un punto de compresión a 1 dB de 30 dBm, un SFDR máximo de 60 dBc y un rango de frecuencia que va desde los 30 MHz hasta los 3 GHz. Este dispositivo, que se encuentra a la salida del conversor DA, cumple una función principal (junto con atenuadores fijos) que es la de adaptar los niveles de la señal eléctrica OFDM a los aceptables por el transmisor óptico, y una función secundaria que es la de controlar la onda reflejada producto de la desadaptación de salida del DAC. En la Figura 2.5 se muestra la respuesta en frecuencia del amplificador ABP0300-01-2730 del fabricante Wenteq Microwave obtenida mediante un barrido en frecuencia desde 0 Hz a 4 GHz realizado con un generador y analizador de señales (ver Figura 2.1). Se observa que la frecuencia de corte (-3 dB) de este amplificador es de 3 GHz.

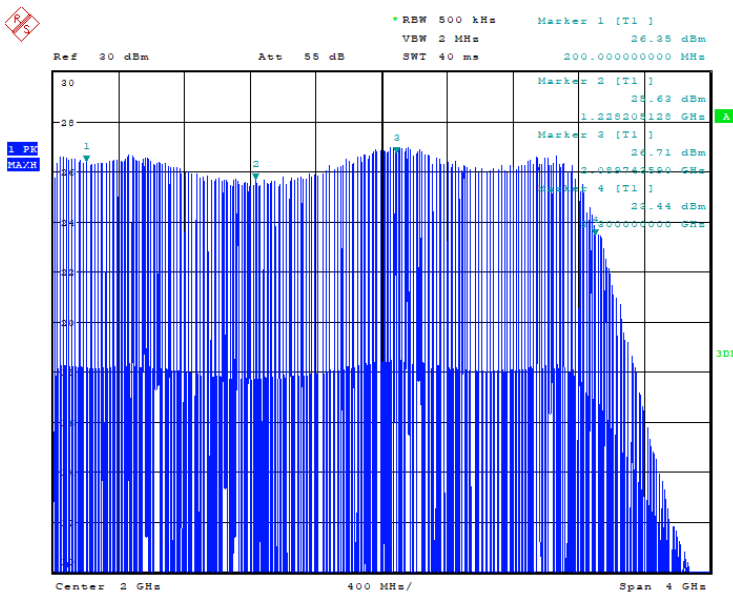


Figura 2.5: Respuesta en frecuencia del amplificador del DAC.

2.3 Etapa óptica

El transmisor óptico A13-Z510 está compuesto por un láser de modulación directa que opera en la longitud de onda de 1550 nm. El receptor óptico A23-Z510 está integrado por un fotodiodo de arseniuro de indio y galio (InGaAs) y un amplificador de bajo ruido. El ancho de banda va desde los 30 MHz hasta los 3 GHz para ambos dispositivos. Este conjunto, de la compañía Optical Zonu, es una solución perfecta para aplicaciones de alto SFDR dentro de una amplia gama de frecuencias.

Para caracterizar la etapa electro-óptica se ha conectado un analizador de redes (HP 8714ET) al transmisor y receptor óptico en un esquema *back-to-back*, tal como se muestra en la Figura 2.1. Posteriormente se han incorporado algunas fibras ópticas de diferentes longitudes para ver el comportamiento del canal. La respuesta en frecuencia de esta etapa se muestra en la Figura 2.6, donde se observa que el sistema comienza a tener una respuesta pasa bajos a partir de los 60 MHz y con dos valles, uno en 700 MHz y el otro 1700 MHz.

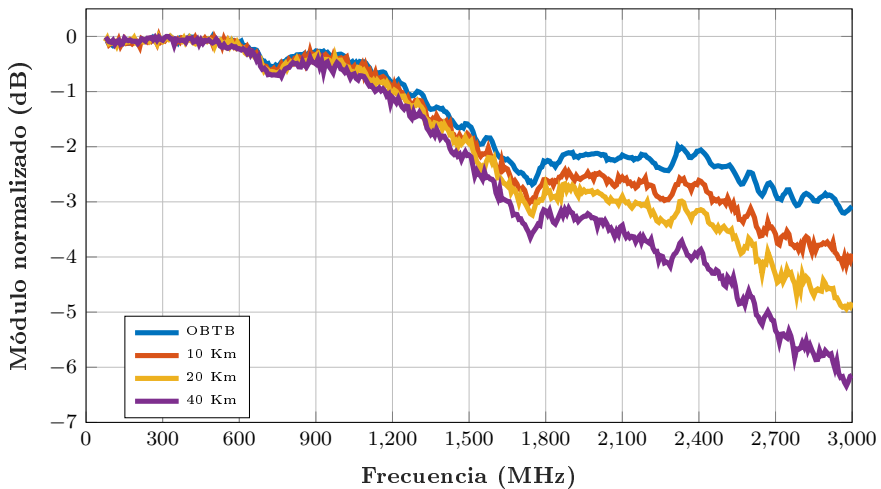


Figura 2.6: Módulo de la respuesta en frecuencia de la etapa electro-óptica para 10/20/40 km de SSMF.

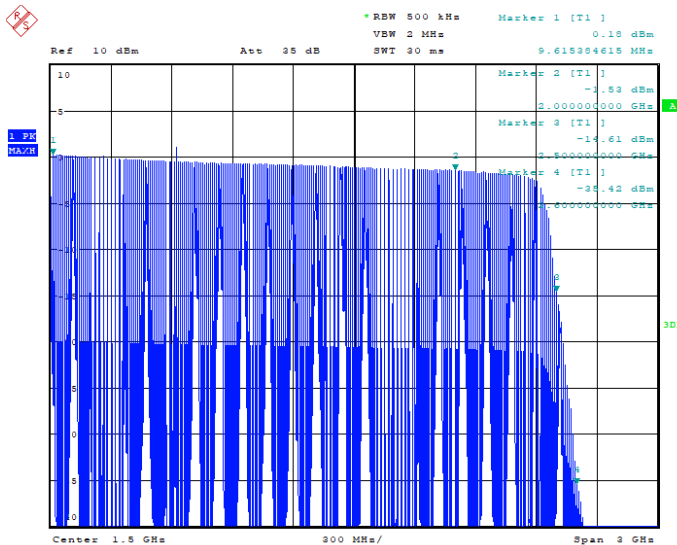
Para el esquema OBTB, el módulo de la respuesta en frecuencia en 2,5 GHz presenta una atenuación de 2,5 dB con respecto a las bajas frecuencias, lo cual se sumará a los casi 10 dB del DAC limitando aún más la carga binaria de las subportadoras de alta frecuencia, tal y como se verá en el capítulo 5. A medida que se incrementa la longitud de la fibra se observa que la pendiente

de atenuación aumenta desde 1.5 GHz en adelante, esto se debe a la dispersión cromática de la fibra y el efecto chirp del DML [39].

En la apartado 1.2 se ha mencionado que existen técnicas para disminuir la potencia óptica media mediante la variación del punto de polarización del modulador del diodo láser [46]. El transmisor óptico A13-Z510 fija internamente el punto de polarización y por lo tanto no se pueden utilizar estas técnicas, ni otras que están relacionadas con la reducción del PAPR o con las distorsiones producto de la no linealidad del diodo.

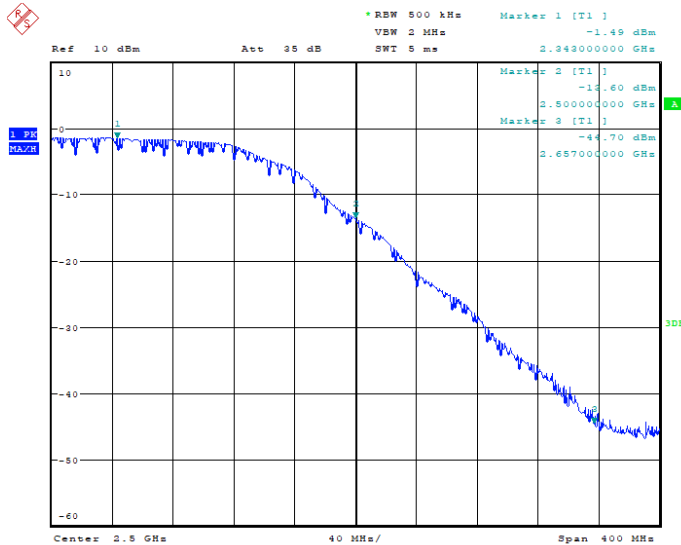
2.4 Filtro Antisolapamiento

El filtro antisolapamiento ha sido diseñado para poder aprovechar al máximo el ancho de banda teórico (0 a 2,5 GHz) y distorsionar lo menos posible la banda de paso. Las especificaciones son las siguientes: banda de paso desde 0 Hz hasta 2.343 MHz, rizado en la banda de paso de 1.5 dB y atenuación en la banda de rechazo (2.656 MHz a 7.000 MHz) de 40 dB. El diseño y la fabricación de este filtro lo ha realizado la compañía TTE y la respuesta en frecuencia se muestra en la Figura 2.7.a.



(a) 0-3000 MHz

Figura 2.7: Respuesta en frecuencia del filtro antisolapamiento.



(b) Banda de transición

Figura 2.7: (Continuación) Respuesta en frecuencia del filtro antisolapamiento.

Este filtro paso bajos personalizado posee una banda de transición muy abrupta (ver Figura 2.7.b) que permite suponer que las primeras 480 subportadoras (desde 0 Hz a 2343 MHz) se encuentra libres de solapamiento o *alias*, para un sistema OFDM con una FFT de 1024 puntos y una frecuencia de muestreo de 5 GHz.

2.5 Conversor AD

Un sistema OFDM para comunicaciones ópticas, necesita para incrementar su velocidad, de un conversor de alta velocidad (mayor F_s) y de gran resolución (mayor SNR). Los conversores analógico digital con entrelazado temporal (*time-interleaved - TI*) [71] cumplen con estos dos requerimientos y, por ello, son los más utilizados para este tipo de aplicaciones. Un estudio realizado acerca de las prestaciones de un gran número de conversores AD desde 1997 hasta la fecha [72], demuestra que la incorporación del entrelazado temporal a las arquitecturas tradicionales (SAR, Flash y Pipe) permite alcanzar tasas de muestreo que van desde algunos GS/s hasta decenas GS/s.

Un TI-ADC es un sistema compuesto por L conversores en paralelo en cual cada conversor toma una muestra a una frecuencia de muestreo menor a la del sistema (generalmente F_s/L) sin tener la necesidad de cumplir con el criterio de Nyquist. Una vez capturadas las L muestras y entrelazadas en el dominio del tiempo, se obtiene una frecuencia de muestreo equivalente que sí cumple con el criterio de Nyquist. Por lo tanto, trabajar con un TI-ADC ideal de L canales es equivalente a trabajar con un ADC que posee una frecuencia de muestreo L veces mayor [71].

Cada conversor que integra un TI-ADC tiene errores dependientes de su tecnología, pero cualquier desajuste entre los canales del TI-ADC hace que aparezcan componentes espurios en el espectro que degradan la relación señal a ruido y distorsión (*signal to noise and distortion ratio - SINAD*). Las principales razones de la degradación del SINAD son tres tipos de desajustes entre canales: diferencias de ganancia, *offset* y temporales [73]. El desajuste temporal nada tiene que ver con el *jitter* del reloj, del cual también padecen este tipo de conversores, sino que está relacionado con la desviación determinística entre el tiempo de muestreo ideal y real (*clock skew*) de cada canal del ADC.

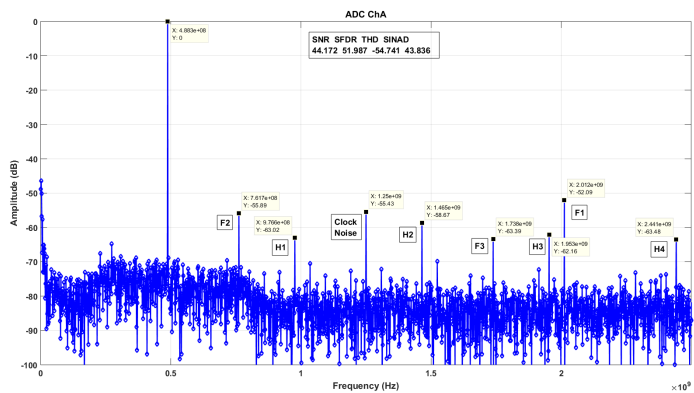
Según lo expuesto en [74], la degradación de la SNR debido a la falta de coincidencia en el *offset* es constante e independientemente de la frecuencia y amplitud de la señal de entrada. También es constante la degradación debido al desajuste de ganancia, pero a diferencia del desajuste de *offset* sí depende de la amplitud de la señal de entrada. En el caso de la falta de coincidencia temporal, la degradación de la SNR se incrementa con la frecuencia de la señal de entrada.

Las causas de los desajustes entre canales en un TI-ADC incluyen: la distribución imperfecta del reloj, la variación en el tamaño de los transistores, las diferencias en los caminos de la señal y los efectos parasitarios. La compensación de estos efectos para los TI-ADCs de uso general ha sido estudiada en [71], y particularizada para el uso en sistemas OFDM en [75, 76]. En estos últimos trabajos se concluye que para SNR altos, como por ejemplo los presentes en sistemas que utilizan fibra óptica, puede aparecer un piso de error (*error floor*) en la curva de BER (*Bit Error Rate*) para un desajuste entre canales mayor al 5%, lo cual podría hacer inviable la utilización de ciertos índices de modulación para un determinado umbral de tasa de error. En el siguiente apartado se presenta el método elaborado para calibrar el TI-ADC EV10AQ190A utilizado en esta tesis. Este conversor se compone de 4 conversores AD de 10 bits que pueden alcanzar una tasa de muestreo de hasta 1,25 GS/s cada uno.

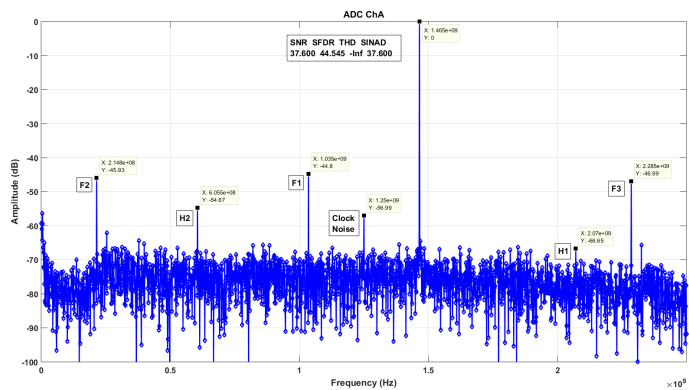
Una característica no deseable de los conversores AD es la generación de espurios, que son dependientes de la frecuencia de muestreo utilizada y de la calibración analógica. A continuación se presentan 3 espectros en la Figura 2.8 correspondientes a señales senoidales de diferentes frecuencias, obtenidos a partir de un generador de señales conectado a un filtro antisolapamiento y este a la entrada del ADC ($F_s = 5$ GHz).

Si se analizan las 3 gráficas se observa que la potencia de los espurios conserva un nivel inferior a los -40 dB y que en 1.25 GHz (frecuencia interna de los conversores) siempre hay un espurio de baja amplitud (-55 dB aproximadamente). En las gráficas con H1, H2, H3 y H4 se indican desde el primer armónico hasta el cuarto. Luego con F1, F2 y F3 se indican los espurios producto de los desajustes de ganancia y temporal entre los 4 canales del TI-ADC. El espurio que aparece en 1.25 GHz está relacionado con el desajuste de *offset* entre los 4 canales y se lo denomina ruido de reloj.

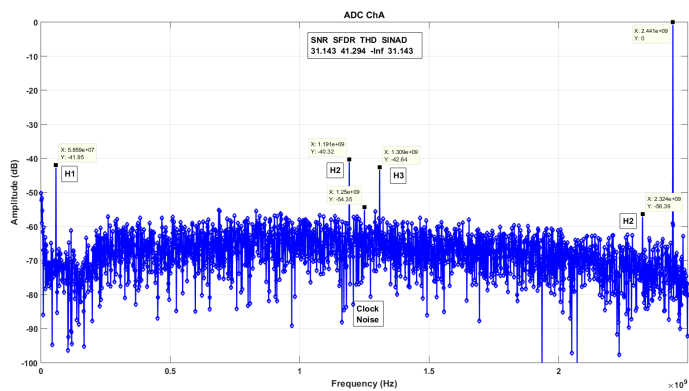
En base a las pruebas realizadas y los valores de SINAD medidos se pueden obtener los valores de ENOB para las diferentes frecuencias según lo descrito en [77]. Estos valores nos hablan de que los ENOB del conversor son bastante menos que los indicados por la resolución (10 bits), que estos varían en función de la frecuencia de la señal capturada. Esta degradación de la SINAD con el aumento de la frecuencia de la señal afectará, como se verá en el capítulo 5, a la carga binaria que se puede asignar a cada portadora de un símbolo OFDM. Además el ruido de reloj afectará a las subportadoras centrales.



(a) 488 MHz ($100 \cdot 5e9/1024$)



(b) 1465 MHz ($300 \cdot 5e9/1024$)



(c) 2441 MHz ($500 \cdot 5e9/1024$)

Figura 2.8: Espectros de los tonos capturados con el ADC EV10AQ190 operando a 5 GS/s.

2.5.1 Calibración analógica

En base a todo lo expuesto anteriormente es necesario realizar la calibración del conversor antes de comenzar a trabajar con él. Para este fin, el conversor EV10AQ190A cuenta con tres ajustes (ganancia, *offset* y temporal) por cada canal, y cada uno de ellos se puede controlar mediante registros internos del conversor que se acceden vía una interfaz digital SPI. El método sugerido por el fabricante se encuentra descrito en [78], para esta tesis, se ha modificado ligeramente basado en otras notas de aplicación y los resultados obtenidos. A continuación se detalla el método utilizado y las mejoras alcanzadas. Este se divide en tres etapas: *offset*, ganancia y temporal.

2.5.1.1 Offset

Se configura el conversor en modo entrelazado (4 canales a 1,25 GS/s desfasados 90°) y se conectan las 4 entradas analógicas a masa mediante cargas de $50\ \Omega$ para asegurar que los 4 canales estén midiendo un mismo valor de tensión (0 V). Los conversores de 10 bits (1024 combinaciones) que integran el TI-ADC trabajan con un sistema de numeración binaria de complemento a uno, razón por la cual, el valor 0 V se representa con el valor entero 511.

Si no existiera un desajuste del *offset* de los 4 conversores y ninguna fuente de ruido, tanto interna como externa, los 4 canales deberían presentar en su salida digital el valor 511. Como en realidad esto no es así, es necesario realizar una medición real y ver cómo se comportan los conversores. Se han capturado un gran número de muestras (16384) por cada canal y se ha calculado el histograma para cada señal en el entorno al valor 511. En la Figura 2.9 se muestran dichos histogramas y de su análisis se pueden concluir que el error de *offset* de cada conversor es diferentes y que la variación entorno al valor medio está acotado a ± 3 cuentas.

Por lo tanto, se propone un método iterativo que intenta llevar el centroide de los 4 histogramas al valor de 511 y que las bandas laterales tengan el menor valor posible. Cada conversor dispone de un registro interno de control de 10 bits que permite modificar el *offset* en 0,08 por cada paso ($\pm 40\text{LSB}/1024$). En cada iteración se calculan dos parámetros: la distancia del centroide al valor de 511 y el valor de las bandas laterales. Con el primero se hace el ajuste grueso de la corrección y con el segundo se hace un ajuste más fino. Después de 10 iteraciones se ha visto que ya no se logran mejoras. En la Figura 2.10 se muestran los histogramas de las 4 señales después de la calibración.

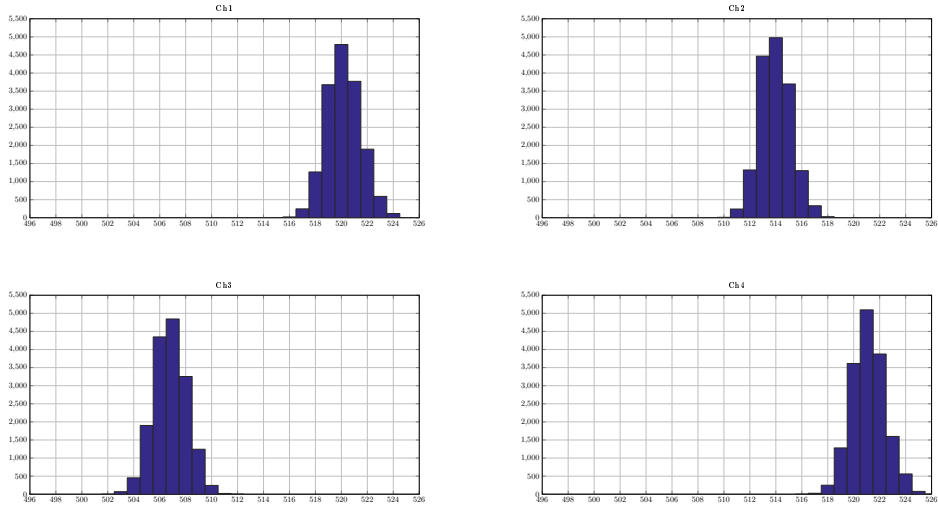


Figura 2.9: Histogramas de los 4 canales con sus entradas a 0 V sin calibración del *offset*.

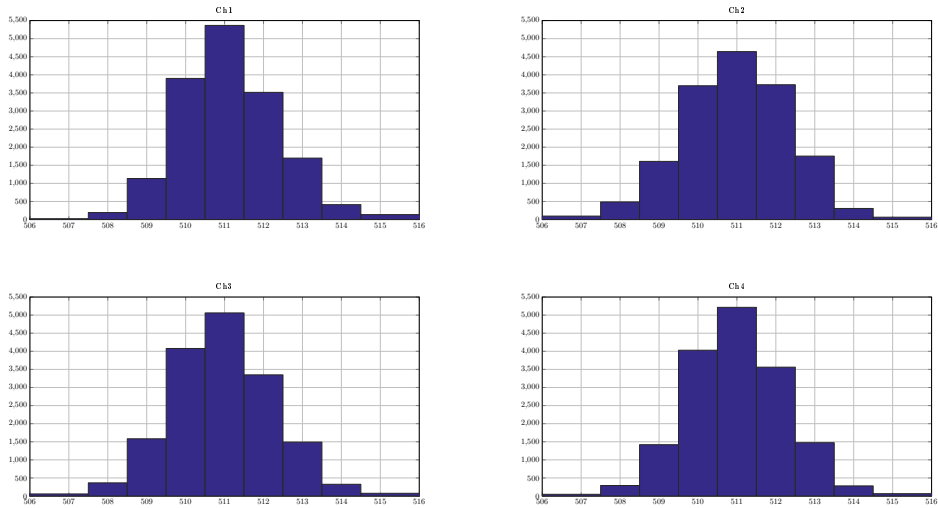


Figura 2.10: Histogramas de los 4 canales con sus entradas a 0 V con calibración de *offset*.

Desde el punto de vista del dominio de la frecuencia, el desajuste de *offset* sólo afecta a los espurios $k \cdot Fs/8$ [73], donde Fs es la frecuencia de muestreo del convertor y $Fs/4$ la frecuencia de muestreo de cada uno de los canales. El espurio que se produce a $Fs/2$ no afecta a la SINAD del convertor. Para verificar la calibración del *offset* y observar el efecto de la misma en el espectro,

se configura el modo de trabajo entrelazado de 4 canales del ADC, en una de sus entradas se inyecta un tono puro de 600.280.761 Hz y se captura un gran número de muestras. En la Figura 2.11 se muestra el módulo del espectro de la señal captura después de la calibración del *offset*. Se observa que los espurios $F_s/4$ y $F_s/2$ disminuyen casi 25 dB y 16 dB, respectivamente, y que los espurios $F_s/8$ y $3F_s/8$ casi no sufren modificaciones; esto último concuerda con la información provista por el fabricante que dice que sólo los espurios $F_s/4$ y $F_s/2$ se verán afectados.

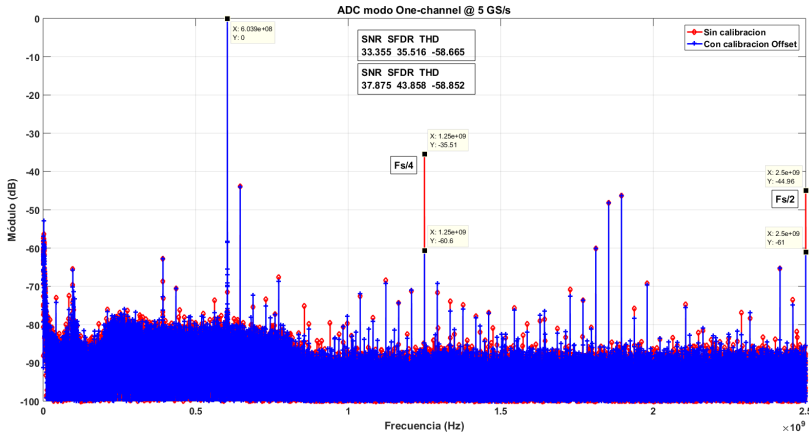


Figura 2.11: Espectro para un tono puro de 600 MHz con y sin calibración de *offset*.

2.5.1.2 Ganancia

Se configura el conversor en modo entrelazado, se inyecta a las 4 entradas analógicas un tono puro de 600.280.761 Hz (F_{in}), se realiza una medición de un gran número de muestras (16384) por canal y se calcula el módulo del espectro de los 4 canales. Lo que se busca es igualar la potencia de la componente fundamental F_{in} en los 4 canales, para ello se toma un canal como referencia y se compensa la ganancia de los restantes canales utilizando los registros de compensación de ganancia. Cada conversor dispone de un registro interno de control de 10 bits que permite modificar el *offset* en 0,02% por cada paso ($\pm 10\%/1024$).

El valor de frecuencia de la señal utilizada para la calibración no es arbitrario, sino que responde a las especificaciones descritas en [78]: $F_{in}/F_s = M/N$. Donde F_s es la frecuencia de muestreo, M es el número de ciclos del tono

de entrada (debe ser impar) y N el número de muestras adquiridas. Se han escogido los siguientes valores $M=1979$, $F_s=5$ GHz y $N=16384$ dando como resultado $F_{in}=603.942.871$ Hz.

Desde el punto de vista del dominio de la frecuencia, el desajuste de ganancia sólo afecta a los espurios $\pm F_{in} + k \cdot F_s/8$ [73]. De todos estos espurios el fabricante informa que los más significativos son $F1$, $F2$ y $F3$, donde $F1=F_s/2 - F_{in}$, $F2=F_s/4 - F_{in}$ y $F3=F_s/2 - F2$, todos ellos afectan al SINAD. Para verificar la calibración de la ganancia y observar el efecto de la misma en el espectro, se configura el modo de trabajo entrelazado de 4 canales del ADC, en una de sus entradas se inyecta un tono puro de 600.280.761 Hz y se captura un gran número de muestras. En la Figura 2.12 se muestra el módulo del espectro de la señal captura y se observa que los espurios $F1$, $F2$ y $F3$ disminuyen casi 12 dB, 8 dB y 3 dB, respectivamente, después de la calibración; el resto del espectro no sufre modificaciones.

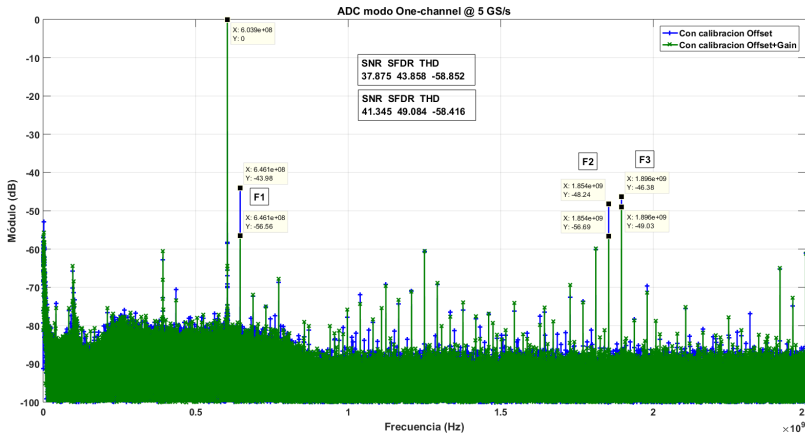


Figura 2.12: Espectro para un tono puro de 600 MHz con y sin calibración de ganancia.

Los valores de frecuencia los espurios a analizar se obtienen a continuación:

$$F1 = 2.500.000.000 \text{ Hz} - 603.942.871 \text{ Hz} = 1.896.057.129 \text{ Hz}$$

$$F2 = 1.250.000.000 \text{ Hz} - 603.942.871 \text{ Hz} = 646.057.129 \text{ Hz}$$

$$F3 = 2.500.000.000 \text{ Hz} - 646.057.129 \text{ Hz} = 1.853.942.871 \text{ Hz}$$

2.5.1.3 Temporal

Desde el punto de vista del dominio de la frecuencia, el desajuste temporal afecta de igual manera que el desajuste de ganancia [73], por lo tanto, el procedimiento para su ensayo es el mismo. Se busca disminuir la potencia de los espurios $F1$, $F2$ y $F3$, para ello se toma un canal como referencia y se comienza a variar el retardo de apertura de los restantes canales utilizando los registros de compensación de fase. Cada conversor dispone de un registro interno de control de 10 bits que permite modificar el retardo de apertura en 30 fs por cada paso (± 15 ps/1024).

Se han realizado pruebas de manera exhaustiva y no se han logrado mejoras significativas con esta calibración.

2.6 Reloj de muestreo

En sistemas reales nunca se puede garantizar que la frecuencia de muestreo en el transmisor y en el receptor sean exactamente iguales, esto es debido a que los osciladores tienen la misma frecuencia nominal pero nunca son iguales. Este desvío en frecuencia de su valor nominal se cuantifica mediante partes por millón (ppm), es un valor que debe especificar el fabricante del oscilador.

La diferencia de frecuencia entre los osciladores que actúan sobre los conversores DA y AD, se traduce en una diferencia en la frecuencia de muestreo entre el transmisor y el receptor. A este suceso se lo denomina desplazamiento de frecuencia de reloj de muestreo (*Sampling Clock Frequency Offset - SCFO*), el cual necesita ser estimado y compensado en el receptor para evitar la degradación de las prestaciones. Este tema ha sido ampliamente estudiado por otros autores y en el capítulo 5 se abordará brevemente.

Para eliminar los efectos que ocasiona el SCFO se ha optado por utilizar la misma fuente de reloj para ambos conversores, y no estrictamente el mismo reloj ya que el ADC requiere una señal de reloj de muestreo cuya frecuencia sea la mitad de F_s mientras que el DAC requiere que esas frecuencias sean iguales. El esquema de generación de relojes de muestreo es el que se muestra en la Figura 2.13, en él se ha colocado un multiplicador de frecuencia por 2 (ZX90-2-36) en una de las salidas de un *splitter* (ZN2PD-63) para duplicar la frecuencia del oscilador del sistema (ASG-6000-U). Los filtros utilizados son para reducir armónicos y disponer así de un reloj de muestreo lo más puro posible. El duplicador de frecuencia tiene muchas pérdidas y hace necesario la utilización de un amplificador (ZX60-5916MA). En la Figura 2.14 se muestra el

módulo del espectro de las señales de reloj de muestreo de ambos conversores, se puede ver que el contenido armónico está 45 dB por debajo de la fundamental.

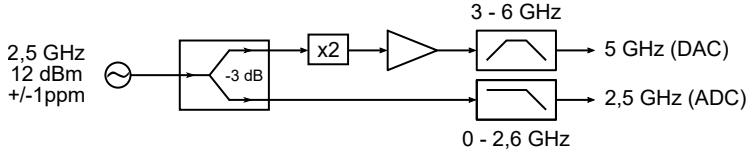
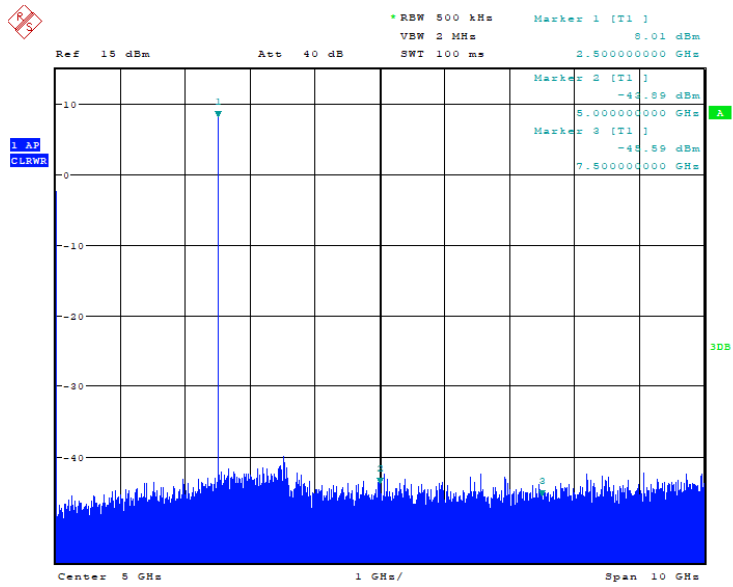


Figura 2.13: Generación de relojes de muestreo para DAC y ADC.

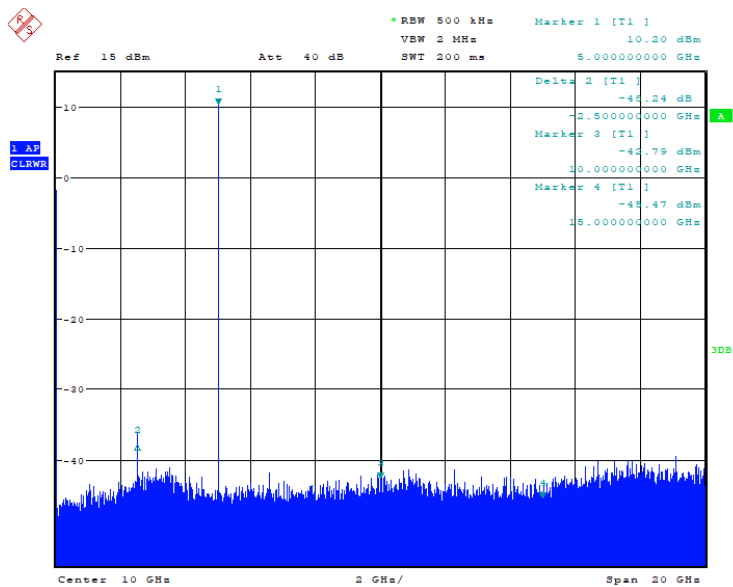
2.7 Conclusiones

El trabajo descrito en este capítulo ha permitido conocer en profundidad cada uno de los elementos *hardware* que integran este módem OFDM para comunicaciones ópticas y conocer desde el principio las limitaciones de dichos componentes que condicionan el diseño final del sistema de comunicación. Algunas pocas subportadoras de muy baja frecuencia deberán ser moduladas con pocos bits debido a la respuesta pasa altos de algunos componentes. Las subportadoras de más alta frecuencia estarán muy afectadas (valores de EVM altos) por la respuesta pasa bajos de todo el conjunto y la generación de espurios de los conversores. Producto del ruido del reloj de muestreo del ADC utilizado, las subportadoras del entorno de 1,25 GHz no podrán utilizar modulaciones altas.

Esta tarea ha requerido un gran trabajo de instrumentación en el laboratorio, el desarrollo de algunos bloques dentro de las FPGAs, varias herramientas y algoritmos realizados en MATLAB y muchas semanas de investigación para profundizar en el funcionamiento de los conversores, su caracterización y puesta en marcha. Todo este trabajo y esfuerzo se verá reflejado en los resultados que se presentarán en el capítulo 5 utilizando los componentes aquí estudiados para diferentes esquemas de trabajo: *back-to-back* eléctrico (EBTB), *back-to-back* óptico (OBTB) y 10/20/40 km de SSMF.



(a) Reloj de 2,5 GHz



(b) Reloj de 5 GHz

Figura 2.14: Espectro de frecuencia de los relojes de muestreo.

Diseño de un módem OFDM para comunicaciones ópticas

Los sistemas de comunicaciones ópticas basados en OFDM utilizan tasas de muestreo de varias Giga muestras por segundo (GS/s), razón por la cual, se hace inviable utilizar una arquitectura monoprocesador para la implementación de cada uno de los bloques que integran el sistema de la Figura 1.3. Por lo tanto, es necesario diseñar algoritmos que admitan una eficiente implementación paralela que permita disminuir la tasa de datos y sean factibles de llevar a cabo en dispositivos con gran capacidad de paralelización, tales como: FPGAs o circuitos integrados de aplicación específica (*Application Specific Integrated Circuit* - ASIC). Convertir algoritmos secuenciales a totalmente paralelos, disminuye de manera muy significativa, la frecuencia de trabajo de los dispositivos, pero requiere de áreas elevadas que en dispositivos FPGA no siempre se encuentran disponibles o encarecen muchísimo su coste. En dispositivos ASIC se dispone de mayores áreas, pero el consumo y la disposición de los metales son críticas en diseños de alta velocidad. Por lo tanto, existe una relación de compromiso entre la tasa de muestreo, el grado de paralelismo y el área utilizada.

Como se explicará más adelante, en el capítulo dedicado a la implementación *hardware*, la tasa de muestreo requerida y los dispositivos utilizados (DAC, ADC y FPGA) para llevar adelante este módem OFDM óptico, determinan que

es necesario procesar 16 muestras en paralelo simultáneamente a una frecuencia de reloj de 312,5 MHz. Este grado de paralelismo hace necesario reducir, tanto como sea posible, la complejidad de los algoritmos de procesamiento digital de la señal que componen el receptor OFDM. Para disminuir el coste computacional es importante seleccionar adecuadamente los algoritmos que permitan un uso eficiente de los recursos *hardware* cuando se paraleliza su diseño y realizar un estudio pormenorizado de la precisión finita de los mismos.

3.1 Transmisor OFDM

El transmisor OFDM desarrollado en esta tesis funciona en tiempo real utilizando un conversor DA de 12 bits a una tasa máxima de 5 GS/s y un dispositivo FPGA Virtex-7. Para reducir los tiempos de desarrollo e implementación, al mismo tiempo que dotar de más flexibilidad al sistema para poder ser utilizado en otras aplicaciones y tareas del grupo de investigación, se ha optado por desarrollar un generador arbitrario de funciones y generar la señal OFDM a transmitir mediante MATLAB.

3.1.1 Generador Arbitrario de Funciones

Este generador está compuesto de tres grandes bloques: almacenamiento de muestras (memoria), control e interfaz de usuario (CIU) y 2 conversores DA. En la memoria se almacenan muestras de 12 bits en formato entero sin signo para cada uno de los conversores. El CIU permite leer, escribir y reproducir las muestras almacenadas a través de un protocolo propietario que trabaja sobre UDP y que permite comunicar al ordenador con el generador arbitrario de funciones (*Arbitrary Waveform Generator* - AWG) mediante un conexión de gigabit ethernet.

El tiempo entre muestra y muestra se determina mediante la frecuencia de un reloj externo que se conecta directamente a la tarjeta del conversor DA. Los conversores utilizados disponen de dos tipos de repuesta al impulso, con y sin retorno a cero (RZ y NRZ), lo que permite generar señales más allá de la banda base.

3.1.2 Generación de trama OFDM

La señal a transmitir es una trama compuesta por tres grandes bloques de símbolos OFDM tal como se ilustra en la Figura 3.1. El primero se corresponde con los símbolos OFDM que se utilizan en el receptor para detectar el comienzo de trama (sincronismo temporal), el segundo con los que se utilizan para estimar la respuesta en frecuencia del canal y, por último, se encuentran los símbolos que transportan la información o datos. A estos dos primeros bloques se los denomina preámbulo y el diseño de los mismos se detalla a continuación en los apartados 3.2.1 y 3.2.3. Los símbolos OFDM de cada uno de los tres bloques están escalados para que tengan la misma potencia.

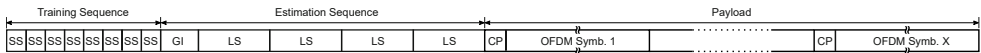


Figura 3.1: Estructura de la trama OFDM propuesta.

La información viaja a través de los símbolos OFDM del *payload* acompañados de su correspondiente prefijo cíclico. Desde el punto de vista meramente teórico el tamaño del símbolo OFDM (N) no afecta a la tasa de información, ni a la eficiencia espectral, pero sí a la potencia de los lóbulos laterales del espectro de la señal transmitida; a medida que aumenta N más rápido decrece la potencia de dichos lóbulos [33]. En la práctica, disponer de más subportadoras permite ajustar de manera más eficiente la modulación utilizada para cada una de ellas, permitiendo una mejora de la tasa de información y de la eficiencia espectral, que en el siguiente apartado se explicará.

La longitud del CP (N_{cp}), desde el punto de vista teórico, se establece teniendo en cuenta la respuesta al impulso y la dispersión cromática del enlace. Ahora bien, desde el punto de vista de la implementación, sería beneficioso que esta longitud sea un múltiplo entero de la cantidad de muestras que se procesan en paralelo dentro de la FPGA. La relación entre el tamaño del símbolo OFDM y su prefijo (N/N_{cp}) es quizás uno de los factores determinantes de la tasa binaria y la eficiencia espectral. Por lo expuesto previamente, es recomendable utilizar el mayor número de subportadoras disponibles y un prefijo cíclico lo más pequeño posible; para este diseño se han escogido los valores de 512 ($N=1024$) y 16 respectivamente.

Los bits de información a transmitir se reparten entre las subportadoras activas de cada símbolo OFDM según el índice y el tipo de modulación utilizado. Como se detalla más adelante en el apartado 3.1.3.2, se ha utilizado una modulación QAM con diferentes índices de modulación, que según la bibliografía actual se conoce como M -QAM. La cantidad de subportadoras activas (N_{usc}) suele

ser inferior al número de subportadoras que pueden transmitir información ($N_{sc} = N/2$ debido a la simetría hermítica) y esto se debe a que, según las características del canal de transmisión, y la respuesta en frecuencia y SNR de los dispositivos analógicos utilizados, la SNR en el receptor puede alcanzar en ciertas zonas del espectro utilizado valores muy bajos que hagan que no merezca la pena enviar información en esas subportadoras.

La generación de la trama OFDM se realiza de manera *off-line* en el ordenador utilizando MATLAB y su *toolbox* de comunicaciones. Primero se genera un vector de muestras en punto flotante escaladas a ± 1 y posteriormente se convierte a punto fijo de 12 bits para que sea compatible con el conversor DA utilizado. Se verá a continuación que se pueden realizar algunas tareas adicionales para mejorar las prestaciones globales del sistema.

3.1.3 Optimización del Transmisor

En este apartado se presentan tres técnicas para mejorar la tasa binaria y la eficiencia espectral del sistema OFDM óptico desde el punto de vista del transmisor teniendo en cuenta las características reales del sistema de transmisión.

3.1.3.1 Recorte/limitación de amplitud (*clipping*)

Las señales OFDM tienen una gran PAPR, lo cual significa que algunos valores son muchos más grandes que la media, y esto provoca que no se aproveche de manera eficiente el rango dinámico de los conversores (DAC/ADC) [42]. Para disminuir este efecto no deseado una de las técnicas más utilizadas es la de *clipping*, que consiste en recortar digitalmente los valores que se alejan mucho de la media de la señal OFDM de manera tal de reducir el rango dinámico a cuantificar y logrando así, reducir el ruido de cuantificación y aumentar la SNR de la señal a transmitir. Para una relación de recorte dada, $\mu = A_{clip}/rms(x[n])$, la señal recortada se puede expresar de la siguiente manera:

$$y[n] = \begin{cases} -A_{clip}, & x[n] < -A_{clip} \\ x[n], & |x[n]| \leq A_{clip} \\ A_{clip}, & x[n] > A_{clip} \end{cases} \quad (3.1)$$

donde $x[n]$ es la señal obtenida después de realizar la IFFT, A_{clip} es el valor del umbral de recorte y $rms(x[n])$ es el valor eficaz de la señal $x[n]$. A continuación,

se realiza el proceso de cuantificación el cual debe cumplir que $\gamma A_{clip} = 2^{b-1} - 1$, donde γ es un factor de escala y b la cantidad de bits del conversor.

El efecto de recortar la amplitud de la señal temporal se puede traducir en una fuente de ruido, al igual que el número de bits de los conversores se traduce en ruido de cuantificación. Para determinar analíticamente la distorsión introducida por el recorte y la conversión, la principal suposición es que las muestras de la señal OFDM en el dominio del tiempo pueden aproximarse como variables aleatorias gaussianas de media cero. La forma de abordar esta problemática es mediante un análisis de la SNR de los elementos que componen la siguiente cadena: recorte, DAC y ADC. En [79] se presentan las ecuaciones de ruido equivalente de cada uno de estos elementos y se concluye que existe una relación de compromiso óptima, entre los errores producto del recorte y los errores de cuantificación, que minimiza el error cuadrático medio (MSE). En [80] se demuestra que la relación de recorte óptima (μ óptimo) es independiente del formato de modulación utilizado y de la longitud de la FFT, siempre que las salidas de la FFT estén ajustadas a la unidad, con lo cual el ruido debido al recorte depende únicamente de μ y el ruido de cuantificación de los bits del conversor.

En la Figura 3.2 se presentan los resultados de la SNR para diferentes relaciones de recorte y bits de cuantificación. Por simplicidad se han elegido la misma cantidad de bits para ambos conversores y se ha marcado en negro el μ óptimo en cada una de las curvas. En nuestro diseño los conversores no tienen la misma cantidad de bits y, además, por cuestiones de fabricación y la tecnología utilizada, los bits efectivos de ambos conversores son inferiores a los bits del tamaño de palabra y varían con el rango de frecuencia utilizado. En base a algunas especificaciones dadas por los fabricantes y varias mediciones realizadas en el laboratorio se puede aproximar la cantidad de bits efectivos equivalentes para todo el rango de frecuencias utilizado a 9 y 7 bits para el DAC y ADC, respectivamente. En la Figura 3.3 se muestra la SNR fruto del recorte y cuantificación de ambos conversores, y se destaca el valor del μ óptimo que se deberá utilizar en el transmisor para minimizar este efecto. Este valor es simplemente un punto de partida para que luego, en la etapa experimental, se realicen variaciones del mismo hasta encontrar el valor óptimo del sistema real, ya que para este análisis hay fenómenos que no se han tenido en cuenta.

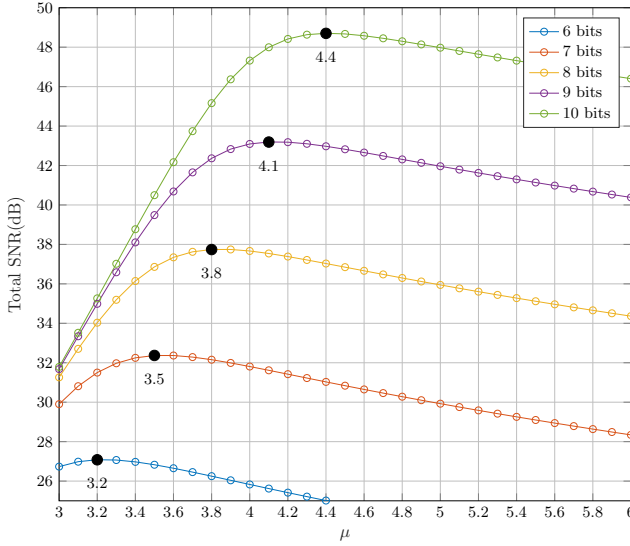


Figura 3.2: Efecto de la relación de recorte para diferentes bits de cuantificación (DAC+ADC).

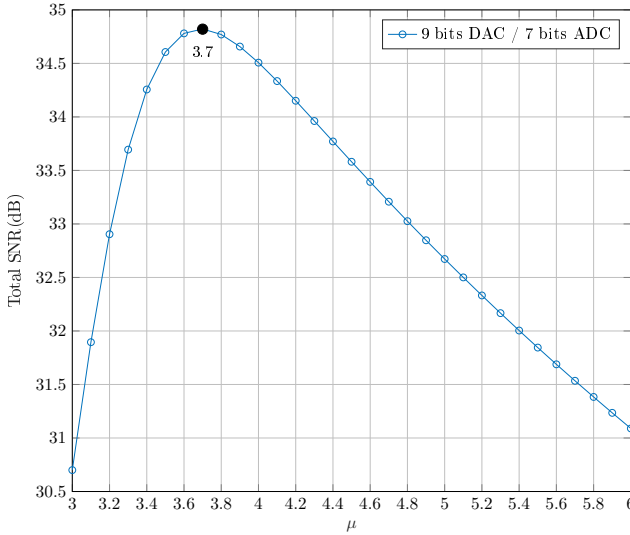


Figura 3.3: Efecto de la relación de recorte utilizando 9 bits DAC y 7 bits ADC.

3.1.3.2 Carga binaria de la subportadoras (Bit Loading)

Como se muestra en la Figura 3.4.a, existe un desvanecimiento de potencia de la señal recibida de aproximadamente 15 dB en alta frecuencia y esto se debe a la excesiva respuesta paso bajo del sistema (DAC, amplificador analógico, filtro antialiasing y ADC). Lo cual implica que, independientemente de cómo y cuáles sean las fuentes de ruido, a medida que aumenta la frecuencia la SNR de las subportadoras irá disminuyendo y la capacidad de transmitir información también. Para mejorar el rendimiento del sistema se pueden aplicar varios algoritmos de carga adaptativa a cada subportadora individual de acuerdo con su SNR ó EVM, existe una relación directa entre estas dos métricas [41]. En [53, 81] se presentan tres algoritmos de carga para sistemas OFDM ópticos de detección directa: carga de bits, carga de potencia, y carga de bits y potencia. Los autores concluyen que el algoritmo de carga de bits (*Bit Loading* - BL) ofrece una buena relación de compromiso entre la capacidad de la señal y la complejidad del *hardware*, lo cual cumple con uno de los objetivos especificados en este trabajo de tesis.

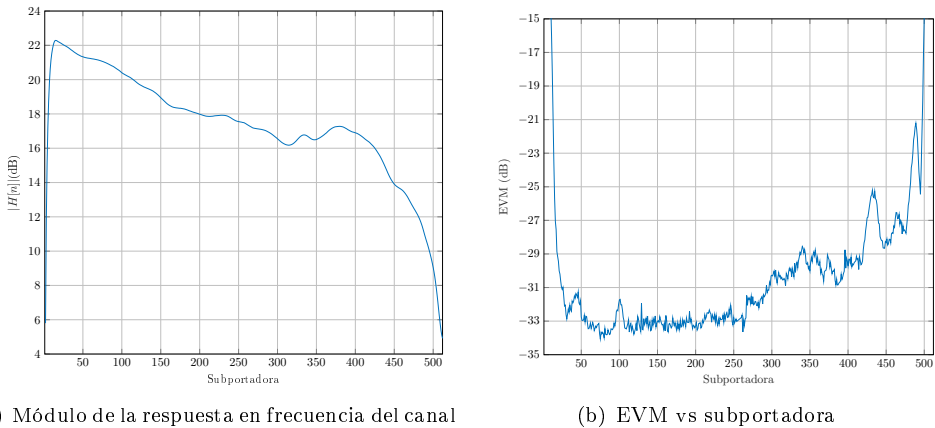


Figura 3.4: Caracterización del módem OFDM para el caso de EBTB.

La estrategia utilizada para establecer el orden de modulación de cada subportadora es la siguiente. Primero se obtienen mediante simulación [41, 82] las curvas de BER en función del EVM para las diferentes modulaciones (ver Figura 3.5), posteriormente se obtiene de manera experimental el valor de EVM para cada subportadora (ver Figura 3.4.b) y, por último, se establece cuál es límite de EVM para cada índice de modulación QAM según un criterio de tasa de error aceptable, en este caso de $3,8 \times 10^{-3}$ para HD-FEC (*Hard-Decision*

Forward Error Correction) [83]. Si analizamos la Figura 3.5 para el caso de las constelaciones rectangulares se observa que para límite del decodificador de errores (*FEC limit*) hay un salto de 6 dB en el valor de EVM. Esto implica que para las subportadoras con valores de EVM dentro de este intervalo, si se les aplica el índice de modulación que se encuentra a la derecha del mismo, las subportadoras generan menos errores de los permitidos, al mismo tiempo que transmitirán menos información. Si se incorporan modulaciones no rectangulares, se puede disminuir este intervalo y cargar con un bit más algunas subportadoras, incrementando de este modo la tasa binaria y mejorando la eficiencia espectral. Nuevamente, esto es un punto de partida que luego de manera experimental se debe ajustar para cada esquema de transmisión.

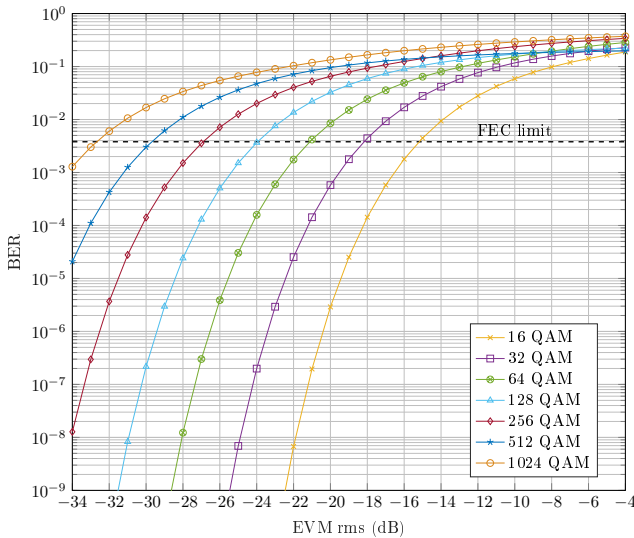


Figura 3.5: BER vs EVM para monoportadora y el marcador para una BER de $3,8 \times 10^{-3}$.

3.1.3.3 Preénfasis

Otra técnica que se suele utilizar para mejorar las prestaciones de un sistema OFDM es el preénfasis digital. Esta técnica, como se muestra a continuación, no provoca ninguna mejora en el sistema objeto de diseño en esta tesis.

En [84-87] se presentan resultados comparativos entre la utilización o no de preénfasis digital en sistemas OFDM ópticos de detección directa, donde se concluye que la utilización de esta herramienta mejora las prestaciones (SNR, EVM, BER). En estos trabajos no se utilizan conversores (AD y DA) comer-

ciales, sino que en su lugar se utilizan generadores de forma de onda arbitraria y osciloscopios digitales trabajando a una frecuencia de muestreo varias veces superior (aproximadamente 3) al ancho de banda de la señal OFDM, lo cual lo aleja bastante de nuestro diseño de bajo coste en tiempo real.

Para entender las ventajas de utilizar instrumentos de laboratorio frente a conversores comerciales es necesario revisar algunos conceptos fundamentales del funcionamiento de estos dispositivos. Comencemos por la respuesta en frecuencia de un convertor digital analógico ideal que se muestra en la Figura 3.6, cuyo comportamiento y ecuaciones se encuentran descritos en [88]. Los conversores DA pueden tener dos tipos de repuesta al impulso distintas: sin retorno a cero (NRZ ó normal hold) o con retorno a cero (RZ). Su función de transferencia es la bien conocida función $\sin(x)/x$ o $\text{sinc}(x)$, con ceros en múltiplos de la frecuencia de muestreo. En ambos casos se observa en la Figura 3.6 que la respuesta en amplitud decrece con el aumento de la frecuencia, con lo cual la potencia de la señal de salida dependerá de la zona de trabajo.

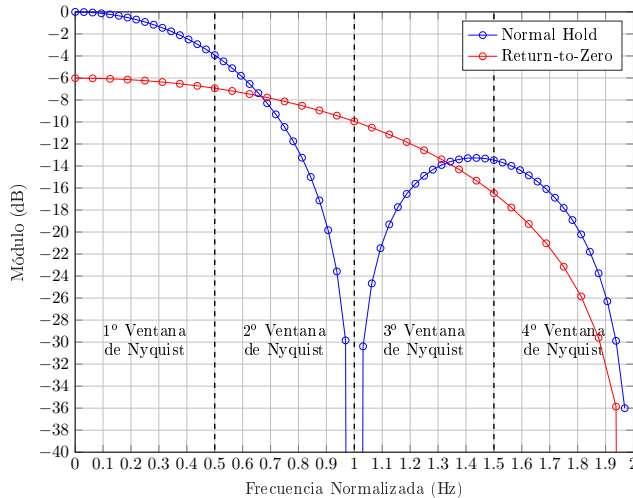


Figura 3.6: Respuesta en frecuencia de un convertor digital analógico ideal.

Otra característica importante de estos conversores es su relación señal a ruido, que como ya se ha dicho en capítulos anteriores, limita la capacidad de transmisión de información. El ruido de un DAC tiene tres posibles fuentes: el circuito, el *jitter* y la cuantificación, siendo esta última la que establece el límite fundamental para la precisión con la que una señal puede ser representada en el dominio digital [88]. Un DAC conserva el error de cuantificación ya presente en sus datos de entrada, sin embargo, puede agregar errores adicionales debido

a varios efectos no ideales que exceden al análisis que aquí se presenta. La SNR de un conversor para una señal con un ancho de banda BW se puede modelar mediante la siguiente ecuación:

$$\text{SNR}_{max} = 6,02b + 1,76 + 10 \log_{10} \left(\frac{F_s}{2 \cdot \text{BW}} \right), \quad (3.2)$$

donde el último término se conoce como la ganancia de sobremuestreo y b se corresponde con la cantidad de bits del conversor. La ecuación 3.2 viene a decir que la SNR de un conversor se puede mejorar utilizando una frecuencia de muestreo superior a dos veces el ancho de banda, y se verá más adelante que muchos artículos mencionados en esta tesis se aprovechan de esto para mejorar las prestaciones del sistema. Por ejemplo, si se sobremuestra por dos ($F_s = 4 \cdot \text{BW}$) se obtiene una mejora de 3 dB de la SNR.

La ecuación 3.2 también modela el ruido de cuantificación en los conversores AD. En [89] se presenta un estudio de cómo el sobremuestreo puede reducir la degradación causada por el *jitter* del conversor AD entrelazado utilizado en sistemas OFDM de alta velocidad. Los autores concluyen que duplicando la frecuencia de muestreo ($F_s = 4 \cdot \text{BW}$) se puede reducir en 3 dB la potencia del ruido debido al *jitter* y que, en presencia de dicho ruido, las subportadoras de alta frecuencia causan más ICI que las de baja frecuencia.

Por último, mencionar que los conversores DA y AD poseen un ancho de banda analógico que, si bien siempre es superior a la frecuencia de Nyquist, modifica la respuesta en amplitud del conversor atenuando un poco las altas frecuencias. A este efecto se podría sumar el ancho de banda del circuito impreso donde se encuentran montados los conversores.

Una vez revisado estos conceptos teóricos, es tiempo de centrarse en la diferencia entre los sistemas que utilizan sobremuestreo [84-87] y los que utilizan todo el ancho de banda disponible ($\text{BW} = Fn$). En la Figura 3.7 se muestra la respuesta de amplitud de un DAC sin retorno a cero en función de la frecuencia (azul), el preénfasis que se debe aplicar a la señal OFDM para el caso de utilizar todo el ancho de banda disponible (magenta) o 1/3 del ancho de banda (verde).

Utilizar sobremuestreo, o lo que es lo mismo que decir un ancho de banda reducido, permite considerar que la respuesta del DAC es prácticamente plana en el ancho de banda utilizado, tal como se ilustra en la Figura 3.7. Esto permite no tener que realizar el preénfasis de la señal. Que la respuesta del conversor

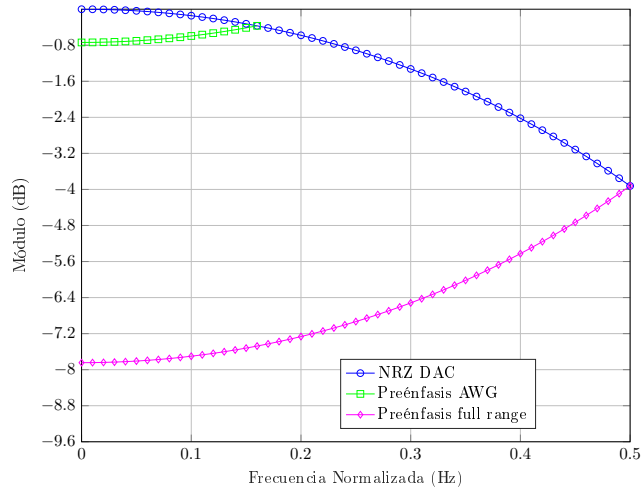


Figura 3.7: Respuesta en frecuencia del DAC en modo NRZ y de los posibles preénfasis.

se pueda considerar plana es una situación ventajosa en comparación con el presente trabajo que necesita al menos realizar un preénfasis de 4 dB. Por otra parte, se logra una mejora de la SNR debido a la ganancia de sobremuestreo (ver ecuación 3.2) en ambos conversores (DA y AD) y los efectos del ancho de banda analógico de ambos conversores se puede considerar despreciable. Todas estas cuestiones, más otros efectos no lineales que para este análisis no se han considerado, hacen que en los sistemas que se utiliza sobremuestreo se pueden alcanzar mayores índices de modulación en un BW varias veces menor al de los conversores. Si la F_s efectiva es menor que la de un sistema sin sobremuestreo esto se traducirá en una mejora de la SE, si por el contrario es igual o superior, además de incrementar la SE también aumentará la tasa binaria.

Si se analiza la Figura 3.7 para el caso de utilizar todo el ancho de banda, la utilización de un preénfasis para corregir al menos la respuesta pasa bajos del DAC implica mantener las más altas frecuencias con la misma potencia y SNR, mientras que las más bajas frecuencias disminuyen su potencia y su SNR. Es por esto que, para el caso de no utilizar sobremuestreo, el preénfasis digital produce una degradación de la SNR de la señal compensada en comparación con la señal no compensada, siendo esta degradación aún mayor para las bajas frecuencias [90]; y todo esto a cambio de una potencia de salida homogénea.

Para ilustrar este análisis se ha transmitido una señal OFDM ($N=1024$) con todas sus subportadoras moduladas con 16 QAM en un esquema *back-to-back*

eléctrico con y sin preénfasis digital del convertor DA. La señal transmitida, recibida y los resultados obtenidos se presentan las Figuras 3.8 y 3.9. El preénfasis digital utilizado es de 6 dB con una función de transferencia $1/\text{sinc}(x)$. Se han escogido dos conjuntos de subportadoras para el análisis, el grupo 1 (G1) compuesto por las subportadoras 40 a 60 y el grupo 2 (G2) compuesto por las subportadoras 440 a 460. Las mediciones de EVM obtenidas en el ordenador muestran que el G1 presenta -33 dB sin preénfasis y -29 dB con preénfasis, mientras que el G2 mantiene el valor en ambos casos. Esto confirma lo antes dicho, este tipo de preénfasis degrada en mayor medida la SNR de las bajas frecuencias que la de las altas. La degradación en 4 dB de la SNR del G1 hace que las subportadoras de este grupo pasen de poder utilizar una modulación 1024-QAM a 256-QAM para una tasa de error de $3,8 \times 10^{-3}$ (ver Figura 3.5) debido a la utilización del preénfasis digital.

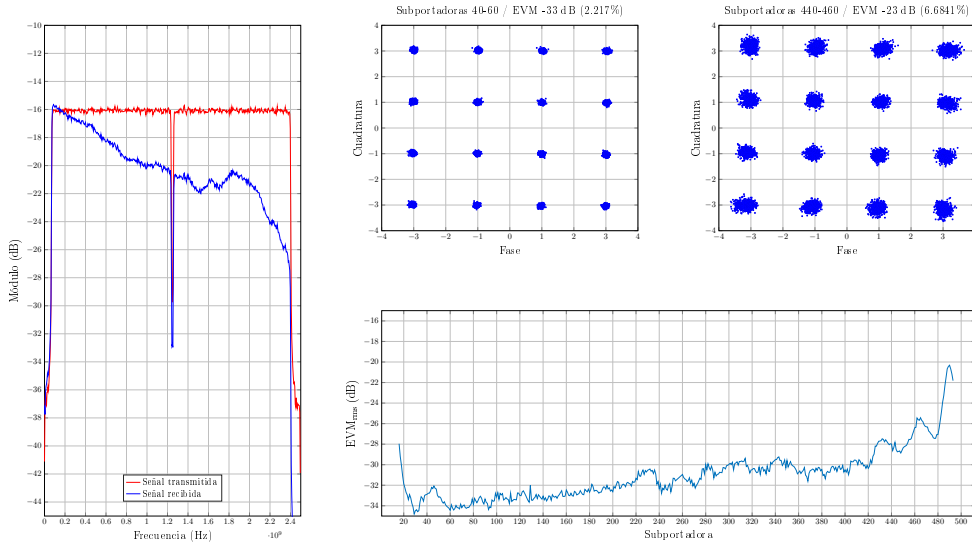


Figura 3.8: Señal OFDM sin preénfasis digital para un esquema EBTB.

Finalizando este apartado, decir que la densidad espectral de potencia de la señal recibida previamente compensada (ver Figura 3.9) no es plana debido a que para el preénfasis utilizado no se han tenido en cuenta los demás efectos del DAC ni el de los restantes elementos (amplificador, filtro, ADC) que componen la cadena analógica del esquema EBTB.

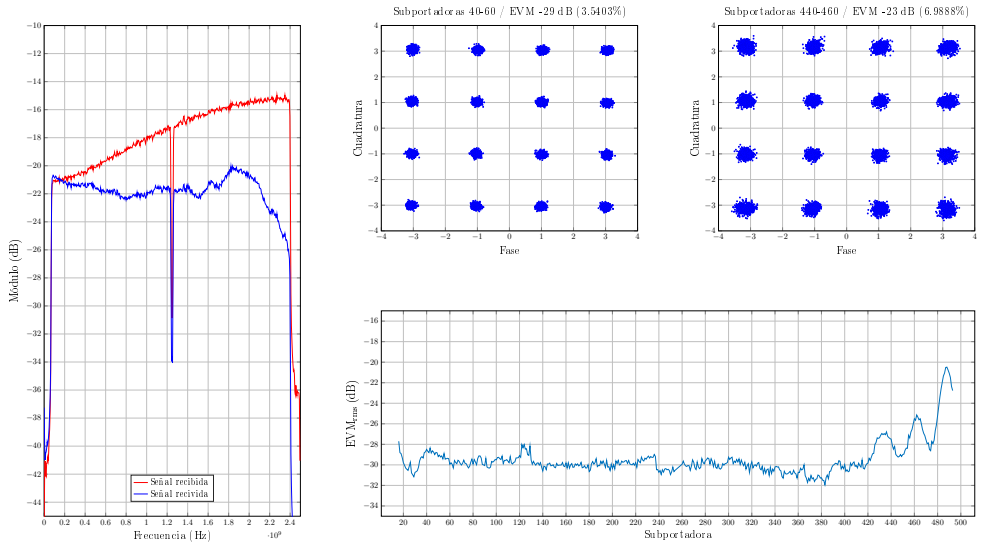


Figura 3.9: Señal OFDM con préenfasis digital para un esquema EBTB.

3.1.4 Modelo de precisión finita

La generación de la trama OFDM y las optimizaciones descritas en los apartados anteriores se realizan en punto flotante de doble precisión (64 bits) para luego ser escaladas a un valor entre ± 1 . Una vez hecho esto, se cuantifica la señal con 12 bits y es aquí donde se introduce el error de cuantificación al que luego se le sumaran el ruido y las no linealidades del convertor DA.

No se ha realizado un diseño de punto fijo de las etapas de procesamiento de la señal del transmisor por simplificar la complejidad del diseño y poder arribar más pronto a la etapa experimental. Además es muy probable que los efectos de precisión finita de estos algoritmos se vieran enmascarados por los bits efectivos de los convertidores y sus alinealidades.

3.1.5 Simulación

Se ha desarrollado un script en MATLAB que permite generar tramas OFDM modificando todos los parámetros y aspectos descritos en esta apartado, así como algunos otros que están vinculados con los algoritmos del receptor (sincronización de tiempo y estimación del canal).

3.2 Receptor OFDM

La complejidad del receptor OFDM es superior a la del transmisor y es por esto que gran parte del trabajo de esta tesis se ha centrado en el diseño e implementación del receptor. El receptor OFDM se compone de las siguientes etapas de procesamiento de la señal: sincronización temporal, eliminación de prefijo cíclico, FFT, estimación y compensación del canal y desmapeo QAM. Dichas etapas se aplican a la señal de banda base recibida y se implementan en el dispositivo FPGA tal como se muestra en el diagrama de bloques de la Figura 3.10.

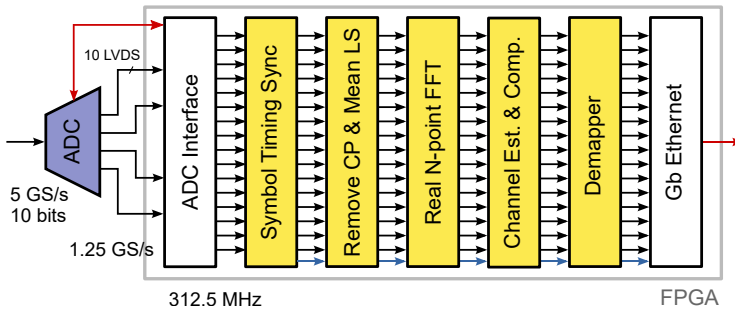


Figura 3.10: Diagrama de bloques DSP del receptor OFDM.

El receptor posee dos bloques adicionales a los bloques de procesamiento digital de la señal. El primero se dedica a calibrar y configurar el ADC, capturar y organizar las muestras que serán enviadas a las etapas de procesamiento digital. El segundo cumple las tareas de control e interfaz de usuario (CIU), y que al igual que en el transmisor, se comunica con el ordenador mediante un interfaz gigabit ethernet.

El receptor OFDM desarrollado en esta tesis funciona en tiempo real utilizando un conversor AD de 10 bits a una tasa máxima de 5 GS/s y un dispositivo FPGA Virtex-7. Las funciones de procesamiento de la señal se realizan dentro de la FPGA y su diseño se detallará en los apartados que vienen a continuación.

La obtención y análisis del EVM y BER se realiza de manera *off-line* en el ordenador utilizando MATLAB. Estas dos mediciones son muy importantes ya que permiten ponderar la calidad y eficiencia de los algoritmos, y como se verá a continuación, determinan cuestiones críticas del diseño.

3.2.1 Sincronizador Temporal

La sincronización temporal es una de las funciones críticas del receptor OFDM, tal como se describió en el apartado 1.1.2. En los últimos años se ha dedicado un gran esfuerzo de investigación a desarrollar algoritmos de sincronización temporal (*Time Synchronization Algorithms* - TSAs) para sistemas OFDM en entornos inalámbricos (ver [91-93] y las referencias que hay en ellos). Algunos de ellos explotan la estructura periódica del prefijo cíclico, otros hacen uso de preámbulos repetitivos y algunos otros utilizan símbolos de entrenamiento diseñados específicamente para mejorar el rendimiento del algoritmo. Una de las técnicas más utilizadas para la sincronización temporal consiste en correlar la señal recibida con una versión demorada de ella misma (autocorrelación), con un preámbulo conocido (correlación cruzada), o una combinación de ambas para detectar el comienzo de la ventana de la FFT [26]. Sin embargo, estos algoritmos no pueden aplicarse directamente a sistemas OFDM ópticos debido a su alta complejidad y a que han sido diseñados para tasas de muestreo bastante inferiores a las utilizadas en OOFDM.

Para alcanzar altas tasas es necesario trabajar con algoritmos con una carga computacional menor y que sean capaces de procesar varias muestras en paralelo. Distintos TSAs se han presentado en [59, 94-98] para sistemas OFDM ópticos. En algunos de ellos ([94, 95]) se explota la periodicidad del prefijo cíclico mientras que en otros ([59, 96-98]) se toma ventaja del diseño de la estructura del preámbulo, de estos últimos la mayoría utiliza la correlación cruzada a excepción de [96].

En esta tesis se ha desarrollado un algoritmo de sincronización de tiempo para sistemas OFDM ópticos con modulación de intensidad y detección directa en tiempo real, basado en la estructura del preámbulo y la correlación cruzada. Se ha tomado como base la idea de emplear un preámbulo repetitivo, como el utilizado en [93], para reducir la complejidad del correlador cruzado y se ha desarrollado una nueva solución para el post-procesamiento de salida que permite reducir el coste *hardware* y mantener buenos resultados de detección. Este TSA realiza una correlación cruzada entre el preámbulo conocido y los datos recibidos sin utilizar multiplicadores, reduciendo de manera significativa el coste computacional del mismo. El uso de un preámbulo repetitivo permite

emplear un correlador cruzado más corto que el utilizado en [59, 97, 98], lo cual es una ventaja desde el punto de vista de complejidad computacional.

A continuación se presenta una versión secuencial del TSA propuesto para comprender su origen y funcionamiento, posteriormente en el apartado 3.2.1.2 se expondrá la versión paralela del mismo.

3.2.1.1 TSA secuencial

El diagrama de bloques del TSA propuesto en esta tesis se muestra en la Figura 3.11. La señal recibida atraviesa una primera etapa de cuantificación que busca reducir el tamaño de palabra con el que deberá trabajar el correlador cruzado, sin perjuicio de la calidad del sincronizador, lo cual se traduce directamente en una reducción de *hardware*. Después la señal de entrada es correlada con un preámbulo conocido y el resultado es procesado por un filtro de media exponencial. Finalmente, se encuentra un bloque de detección de umbral que busca encontrar el pico máximo de la correlación a la salida del filtro, para luego poder indicar el comienzo de la ventana de la FFT para los símbolos OFDM recibidos.

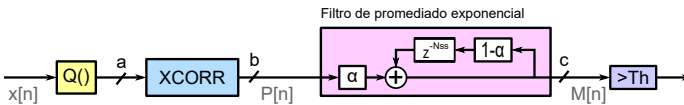


Figura 3.11: Diagrama de bloques del algoritmo de sincronización de tiempo propuesto.

Para llevar adelante esta tarea se ha diseñado un preámbulo (ver Figura 3.12) que se compone de 8 símbolos cortos (*Short Symbols* - SSs) idénticos de N_{ss} muestras cada uno y 4 símbolos largos (*Long Symbols* - LSs) idénticos de N_{ls} muestras cada uno, precedidos por un intervalo de guarda (*Guard Interval* - GI) compuesto por las $2N_{cp}$ últimas muestras del último LS. Donde N_{cp} es el tamaño del CP, N_{ss} es el tamaño de un SS y N es el tamaño de la IFFT. Esta estructura es similar a la utilizada por el estándar WLAN IEEE 802.11 a/g [18, 19]. La parte repetitiva del preámbulo es útil para detectar la presencia de señal, estimar el comienzo de los símbolos de datos OFDM, ayuda a manejar la distorsión en las primeras muestras del preámbulo introducido por el control automático de ganancia (*Automatic Gain Control* - AGC) durante su tiempo de establecimiento, y estimar el desplazamiento en frecuencia de las portadoras (*Carrier Frequency Offset* - CFO). Por último, decir que la estimación de canal se logra utilizando los símbolos largos.

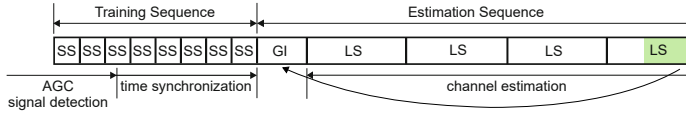


Figura 3.12: Preámbulo propuesto.

Esta parte repetitiva del preámbulo, también llamada secuencia de entrenamiento (*Training Sequence* - TS) a lo largo de este artículo, se genera mediante la modulación por desplazamiento de fase en cuadratura (*Quadrature Phase-Shift Keying* - QPSK) de una de cada 8 subportadoras, mientras que las subportadoras restantes se rellenan con ceros, de un símbolo OFDM cuya longitud es N_{ts} . Por otra parte, la señal de datos OFDM (real) se obtiene generando subportadoras que posean simetría hermitiana alrededor de la subportadora de continua (DC) lo que implica que, sólo $N/2 - 1$ de N subportadoras totales pueden transportar datos debido a que la portadora de DC se utiliza para determinar el punto de polarización del láser. En el dominio del tiempo, TS tiene una estructura con 8 SS como la que se muestra a continuación:

$$TS = [A A A A A A A A], \quad (3.3)$$

donde A es una secuencia de valor real de la longitud $N_{ts}/8$, generado como se ha comentado anteriormente. N_{ts} es el tamaño de la secuencia de entrenamiento. En las mediciones experimentales vamos a utilizar valores de $N_{ts} = 256$ y $N_{ss} = 32$ porque facilitan la implementación paralela del algoritmo, como se discutirá en el apartado 3.2.1.2.

Desde el punto de vista de la implementación, un correlador cruzado es un bloque con una alta complejidad ya que requiere un gran número de multiplicadores y sumadores para procesar datos a varias GS/s y, por lo tanto, se debe optimizar su diseño para reducir su coste computacional. Para alcanzar este objetivo y aprovechando que los valores de SS son conocidos a priori por el receptor, se puede utilizar un esquema sin multiplicadores (ver [93]) donde los valores de SS están representados por su bit de signo y la señal de entrada $x[n]$ es cuantificada con la menor cantidad de bits posible sin perder de vista la eficiencia del sincronizador. De esta manera la correlación cruzada puede ser expresada como en la ecuación 3.4, donde $\text{sgn}()$ es la función de signo. Esta solución permite reemplazar los multiplicadores por una estructura de sumador en forma de árbol gobernada por el signo del SS. Se demostrará más adelante que también es posible cuantificar la señal de entrada con sólo un bit. Un enfoque similar también se ha empleado en [59, 99], donde los multiplicadores han

sido sustituidos por los compuertas lógicas XNOR y a continuación se utiliza un sumador en árbol, esta solución no tiene en cuenta que el SS es conocido por el receptor y necesita más recursos de *hardware*.

$$P[n] = \sum_{m=0}^{N_{ss}-1} Q(x[n+m]) \cdot \text{sgn}(SS[m]) \tag{3.4}$$

La implementación del correlador cruzado como un árbol de sumadores sólo necesita $N_{ss} - 1$ sumadores reales, mientras que la implementación tradicional requiere N_{ss} multiplicadores reales y $N_{ss} - 1$ sumadores reales que trabajan con más bits que los que se utilizan en la ecuación 3.4. Por ejemplo, si usamos $N_{ss} = 32$ y un SS con estos signos $\{+1, +1, +1, +1, +1, -1, -1, +1, +1, +1, +1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, +1, +1, +1, -1, -1, -1, +1, -1, -1, -1\}$ el correlador cruzado se implementa como se muestra en la Figura 3.13, donde el recuadro representa los 4 primeros términos de la correlación cruzada calculada como la ecuación 3.5. Si reagrupamos estos 4 términos para implementar un sumador en árbol de 2 etapas, obtenemos la ecuación 3.6.

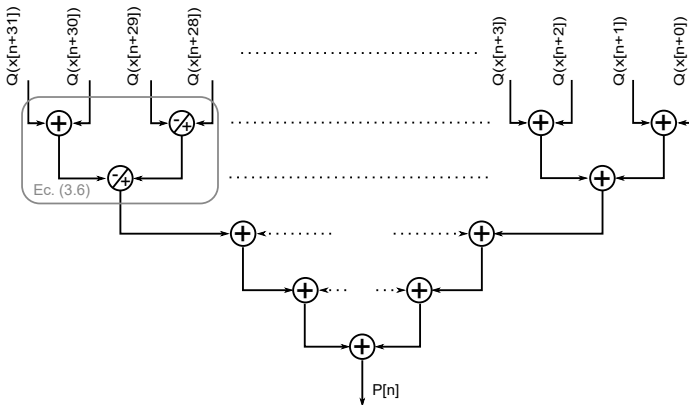


Figura 3.13: Diagrama de bloques del correlador cruzado.

$$Q(x[n+28]) - Q(x[n+29]) - Q(x[n+30]) - Q(x[n+31]) \tag{3.5}$$

$$\{Q(x[n+28]) - Q(x[n+29])\} - \{Q(x[n+30]) + Q(x[n+31])\} \tag{3.6}$$

A la salida del correlador cruzado aparecen unos picos periódicos cuando TS está presente en su entrada, tal como se muestra en color verde en la Figura 3.14. En caso de SNR bajos, los picos se ven un poco deformados y de menor amplitud lo cual dificulta su detección y en casos extremos se asemejan al ruido. Para mejorar la calidad de detección del TSA, la señal correlada se procesa con un filtro de promedio exponencial para resaltar los picos y reducir el ruido de fondo, tal como se describe en la ecuación 3.7. El filtro también evita falsas detecciones cuando no se encuentra presente la secuencia de entrenamiento y el ruido de fondo es alto. Además, gracias a este filtro promediador, una vez que la TS haya finalizado, los picos en la salida se desvanecen rápidamente y de esta manera se simplifica la implementación de un detector de umbral para encontrar el último pico. Mediante la elección del valor de $\alpha=0,5$, ambos productos del filtro de media pueden ser implementadas por una operación de desplazamiento de bits sin coste alguno de *hardware*.

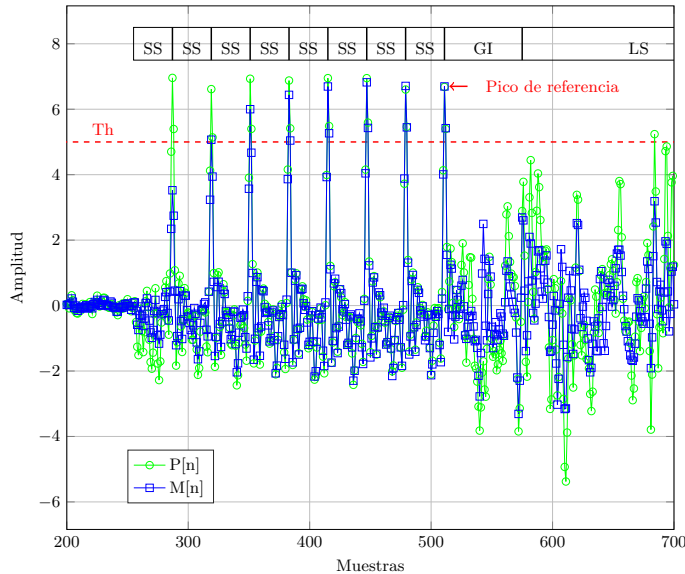


Figura 3.14: Magnitud de la salida del correlador cruzado ($P[n]$) y filtro ($M[n]$) durante la parte repetitiva del preámbulo y en el comienzo del intervalo de guarda.

$$M[n] = \alpha \cdot P[n] + (1 - \alpha) \cdot M[n - N_{ss}] \quad (3.7)$$

Como se observa en la Figura 3.14, la salida del correlador cruzado promediada tiene 8 picos principales, cada uno coincidiendo con la presencia de la última muestra del SS en la entrada del correlador cruzado. El último pico se

utiliza como referencia para la sincronización de tiempo: una vez que el TS ha ingresado al correlador cruzado, se espera un pico cada N_{ss} muestras; cuando desaparece significa que el GI ha comenzado y, por lo tanto, se ha encontrado el último pico. Una vez que se encuentra su ubicación, se utiliza para seleccionar las $2N_{cp} + 4N_{ls}$ muestras de $x[n]$ que corresponden a la última etapa del preámbulo, a continuación, los LS se emplean para estimar la respuesta del canal. El pico de referencia se puede detectar estableciendo un valor de umbral en la salida del filtro y usando una pequeña lógica de control; la selección del umbral se determina de igual manera que en [93].

3.2.1.2 TSA paralelo

Las ecuaciones 3.4 y 3.7 corresponden a un sistema de sincronización temporal de una sola entrada y una única salida (SISO). Para obtener una estructura de procesamiento en paralelo se lo debe convertir en un sistema de múltiples entradas y salidas (MIMO) utilizando la técnica *look-ahead* que se describe en [100]. La ecuación 3.8 describe un sistema de correlación cruzada paralelo con N_p entradas/salidas por ciclo de reloj, donde k denota el ciclo de reloj. El filtro promediador paralelo se describe en la ecuación 3.9.

Para implementar las ecuaciones 3.8 y 3.9 son necesarios $N_p(N_{ss} - 1)$ y N_p sumadores, respectivamente. Para nuestro propósito, es necesario detectar los picos (estableciendo un valor de umbral) a la salida de los N_p filtros de promediado exponencial y determinar cuál es el último. Para llevar a cabo esta tarea se necesitan N_p comparadores, un decodificador con prioridad de N_p entradas y una máquina de estados.

$$\begin{aligned}
 P[kN_p + 0] &= \sum_{m=0}^{N_{ss}-1} Q(x[kN_p + 0 + m])sgn(SS[m]) \\
 P[kN_p + 1] &= \sum_{m=0}^{N_{ss}-1} Q(x[kN_p + 1 + m])sgn(SS[m]) \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 P[kN_p + N_p - 1] &= \sum_{m=0}^{N_{ss}-1} Q(x[kN_p + N_p - 1 + m])sgn(SS[m])
 \end{aligned} \tag{3.8}$$

$$\begin{aligned}
 M[kN_p + 0] &= \alpha P[kN_p + 0] + (1 - \alpha)M[kN_p + 0 - N_{ss}] \\
 M[kN_p + 1] &= \alpha P[kN_p + 1] + (1 - \alpha)M[kN_p + 1 - N_{ss}] \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 M[kN_p + N_p - 1] &= \alpha P[kN_p + N_p - 1] \\
 &\quad + (1 - \alpha)M[kN_p + N_p - 1 - N_{ss}]
 \end{aligned}
 \tag{3.9}$$

La implementación paralela del TSA y el detalle del flujo de procesamiento de la señal se muestra en la Figura 3.15 con $N_p=16$ y $N_{ss}=32$. Este valor de N_{ss} ha sido elegido como una relación de compromiso entre las prestaciones y la complejidad *hardware* del algoritmo, para simplificar la arquitectura paralela del filtro.

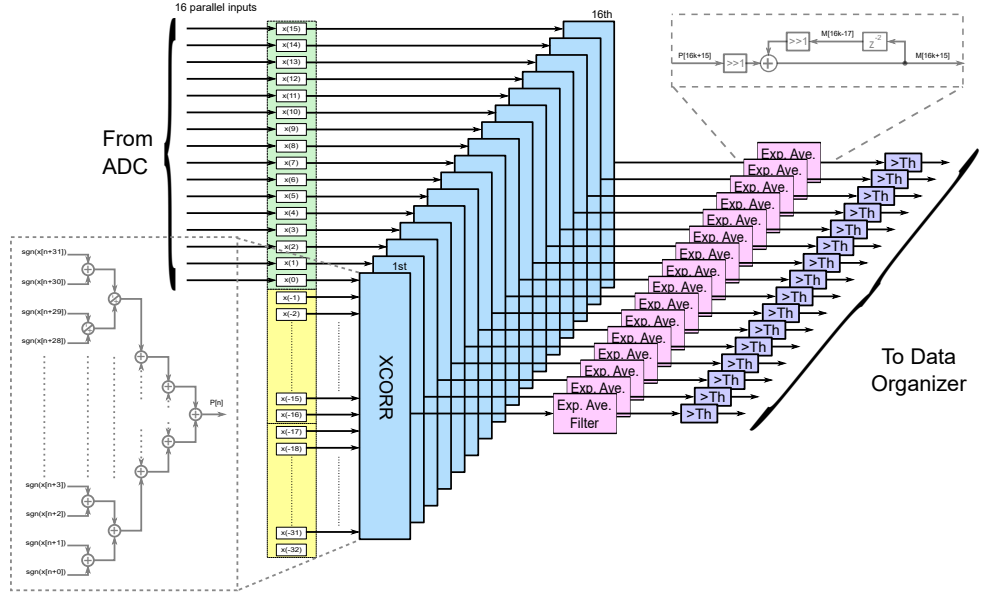


Figura 3.15: Diagrama de bloques del TSA paralelo.

Si $N_{ss} < N_p$, existen largos lazos de retroalimentación en el filtro recursivo paralelo; en este caso, la salida del filtro depende de una salida anterior que se calcula en paralelo. De esta manera se generan caminos combinatoriales entre salidas que, por lo general, son los caminos críticos que limitan la frecuencia máxima de operación del algoritmo. Por otro lado, si $N_{ss} > N_p$ pero N_{ss} no es un múltiplo entero de N_p , se obtiene una estructura *hardware* irregular que

introduce retrasos de enrutamiento y también limita la frecuencia de funcionamiento. Sin embargo, se pueden evitar estos problemas si N_{ss} es elegido como un múltiplo entero de N_p . En este caso, cada salida sólo depende de sí misma retrasada N_{ss}/N_p muestras; por lo tanto, el camino crítico no se incrementa, la estructura de *hardware* se vuelve regular y el filtro paralelo puede implementarse como N_p filtros independientes de una sola entrada/salida. Por ejemplo, en nuestro caso donde $N_p = 16$, un retraso de un ciclo de reloj en el camino de la señal implica un retraso total de 16 muestras. Por lo tanto, para obtener el término $M[16k-32]$ de la ecuación 3.9 tenemos que retrasar la muestra $M[16k]$ dos veces en lugar de 32. Las muestras $M[16k-32]$ y $M[16k]$ son salidas del mismo filtro, esto evita dependencias entre las N_p salidas del filtro paralelo.

3.2.1.3 Modelo de precisión finita

Se ha desarrollado un modelo de punto fijo del TSA utilizando MATLAB en donde las señales se cuantifican con los siguientes números de bits: a para los datos de entrada al correlador, b para los datos de salida del correlador y c para los datos de salida del filtro, tal como se muestra en la Figura 3.11. Como el filtro es un promediador, c y b pueden ser iguales sin dar lugar a posibles desbordamientos. El valor de b depende a y de la cantidad de muestras que se correlacionan (N_{ss}), tal cual como se muestra a en la ecuación 3.10.

$$b = a + \log_2(N_{ss}) \quad (3.10)$$

Las prestaciones del TSA con diferentes valores de cuantificación han sido evaluadas por simulación, y más tarde estos resultados se han validado en un experimento real. Concretamente 10^4 preámbulos han sido generados y transmitidos a través de un canal con ruido gaussiano blanco aditivo (AWGN). Después en el receptor, se ha calculado la probabilidad de una detección temporal correcta (*Probability of Correct Time Detection* - PCTD) definida de la siguiente manera:

$$\text{PCTD} = \frac{\text{Número de detecciones temporales correctas}}{\text{Número total detecciones temporales adquiridas}}, \quad (3.11)$$

donde la detección de tiempo se ha considerado correcta cuando el TSA detecta un pico en la última posición de la TS, o una muestra antes o después.

Los resultados de la simulación se muestran en la Figura 3.16. Se observa claramente que la PCTD de este método no depende de a para valores de

SNR > 3 dB. Por ejemplo, para obtener una BER de 10^{-2} utilizando una modulación QPSK es suficiente con alcanzar un valor de SNR = 3,6 dB. Para estas condiciones, el TSA propuesto estaría trabajando prácticamente libre de errores (PCTD ≈ 1) si se escogen los siguientes tamaños de palabra: $a=1$ y $b=c=6$. En sistemas ópticos los valores de SNR suelen ser bastante mejores. Como el coste *hardware* del algoritmo tiene una fuerte dependencia con los tamaños de palabra utilizados, estos valores pequeños permiten obtener una implementación *hardware* con un bajo coste computacional.

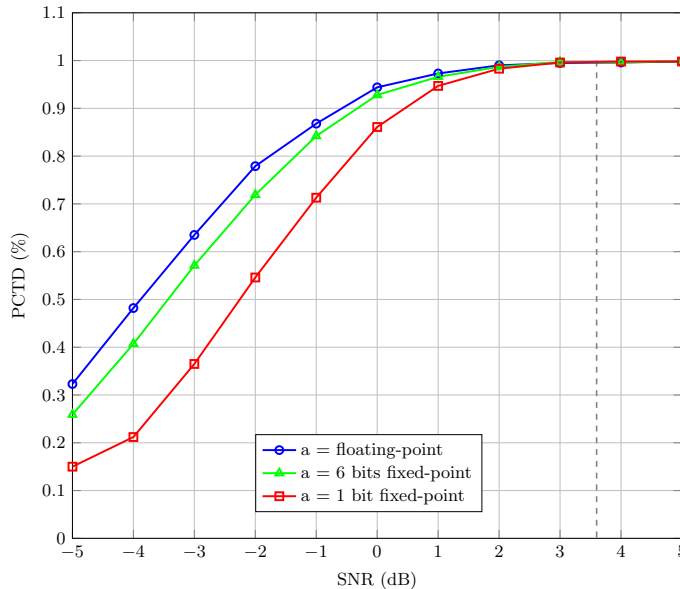


Figura 3.16: Curvas de PCTD para el TSA propuesto ($N_{ss}=32$) versus SNR para diferentes esquemas de cuantificación de la señal recibida y el marcador para una BER de 10^{-2} .

Se ha tomado un valor de BER de 10^{-2} como referencia porque comúnmente se considera el umbral para obtener una transmisión libre de errores cuando se utiliza una SD-FEC (*Soft-Decision Forward Error Correction*) y un 20% de redundancia [101]; por otro lado, en caso de que se empleara una HD-FEC con un 7% de redundancia, el umbral de FEC es de $3,8 \times 10^{-3}$ [83], lo que corresponde a una SNR de 4,9 dB.

3.2.1.4 Comparación de complejidad

A continuación, en la Tabla 3.1, se presenta la complejidad computacional de seis algoritmos de sincronización temporal. Estos TSAs se utilizan en receptores OFDM ópticos y todos ellos procesan N_p muestras en paralelo. En el afán de realizar una comparación más justa de los diferentes algoritmos se han recalculado, sólo en los casos necesarios, los recursos utilizados para la sincronización temporal en un sistema de detección directa. Por ejemplo, los trabajos presentados en [96, 97] se diseñaron para señales OFDM complejas presentes en los sistemas de detección coherente. Las referencias [94, 95] se basan en el cálculo de múltiples diferencias entre la señal de entrada y una versión demorada de la misma, luego un promediado y, por último, un inventariado gaussiano del CP, [96] se basa en la autocorrelación del TS, y [59, 97, 98] se basan en la correlación cruzada del TS.

Tabla 3.1: Comparación de complejidad de TSAs

Algoritmo	Basado en CP/TS	Correlación	Entrada/TS bits	Multiplicadores	Sumadores	XNORs	Max/Th
[94, 95]	CP	-	8/-	$2N_p + N_{cp} + N$	$2(N_p + N_{cp} + N)$	0	Max, Th
[96]	TS	Auto	5/-	2	2	0	Max
[97]	TS	Cruzada	7/1	0	$N_p(N_{ts} - 1)$	0	Th
[59]	TS	Cruzada	1/1	0	$N_p(N_{ts} - 1)$	$N_p(N_{ts})$	Th
[98]	TS	Cruzada	1/1	0	$N_p(N_{ts} - 1) + 2$	$N_p(N_{ts})$	2Th
Propuesto	TS	Cruzada	1/1	0	$N_p(N_{ss})$	0	Th

Los algoritmos que utilizan multiplicadores ([94-96]) necesitan, además, encontrar un máximo (Max) para determinar la posición correcta del inicio del símbolo de datos, lo cual es más costoso computacionalmente hablando, que el uso de una detección de umbral (Th). Los algoritmos basados en la correlación cruzada ([59, 97, 98]) se han implementado sin multiplicadores pero utilizan una compuerta XNOR para multiplicar el signo de la TS con el signo de la señal recibida. En [59, 98] el ancho de palabra medio de los sumadores es más pequeño que los de [97] debido a que la salida del correlador es de 1 bit. El algoritmo aquí propuesto tiene una complejidad menor que el resto debido a que sólo necesita N_{ss} sumadores, donde $N_{ss} \ll N_{ts}$. También se beneficia de no utilizar compuertas XNOR para implementar el correlador, ya que en su lugar utiliza un sumador en árbol cableado. El pequeño número y tamaño de los sumadores en este TSA hace que la latencia sea inferior a la de los otros algoritmos.

El algoritmo propuesto en [96] hace uso de un preámbulo repetitivo de tamaño $8N$ basado en la autocorrelación y, además, necesita un demultiplexado especial de la señal recibida para poder ser procesada, lo que incrementa su coste computacional. En [97] se genera una secuencia de entrenamiento con valores seleccionados al azar del conjunto de $\{-1,+1\}$ y en el lado del receptor los datos recibidos son correlacionados con la secuencia de entrenamiento. Esta operación de correlación se implementa mediante sumas y restas, sin utilizar multiplicadores, lo cual permite reducir la utilización de recursos *hardware*. En [59, 98] el signo de los datos recibidos está correlacionado con el signo de la secuencia de entrenamiento, por lo que las multiplicaciones se reemplazan por funciones lógicas que sólo multiplican los signos (compuertas XNOR). En estos algoritmos, el tamaño del TS es igual a $N_{cp} + N$.

La generación de la TS utilizada por el TSA propuesto aquí se ha descrito anteriormente. En el receptor los datos recibidos son cuantificados y luego correlacionados con el signo de la TS; esta correlación se implementa usando sólo sumas y restas. En el apartado anterior se ha demostrado que basta cuantificar el preámbulo de la señal recibida con 1 bits para obtener un buen rendimiento en el caso de una BER de 10^{-2} ; esta reducción en el número de bits de entrada al algoritmo reduce drásticamente el tamaño de los sumadores/restadores. Por último, decir que la cantidad de sumadores/restadores utilizados depende de N_p y N_{ss} .

Los trabajos presentados en Tabla 3.1 se pueden clasificar en dos grupos dependiendo del uso o no de multiplicadores. Este primer grupo [94-96], además de usar multiplicadores necesitan encontrar un máximo para determinar el inicio de la ventana FFT, esto último desde el punto de vista computacional tiene un coste mayor que usar una detección por umbral. Los del segundo grupo se basan en la correlación cruzada implementada sin multiplicadores. En [59, 98] el promedio del ancho de palabra de los sumadores es menor que en [97], esto se debe al hecho de que mientras que [59, 98] trabajan con cuantificación de 1 bit para las señales de entrada y referencia, en [97] la señal de entrada se cuantifica con 7 bits.

Para terminar con este análisis, decir que el algoritmo diseñado en el marco de esta tesis tiene menor complejidad que el resto de trabajos presentados en esta sección gracias a que solo necesita N_{ss} sumadores por entrada ($N_{ss} \ll N_{ts}$), al uso de sólo el bit de signo tanto de la TS transmitida como recibida y al uso de detección por umbral. También se beneficia del uso de un sumador en forma de árbol en lugar de usar multiplicadores XNOR. Por ejemplo, con $N=256$, $N_{cp}=32$ y $N_p=16$, el TSA descrito en [98] necesita 4.624 sumadores reales, 4.608 XNOR y dos detectores de umbral. El algoritmo presentado aquí

($N_{ss}=32$) necesita 512 sumadores reales y un detector de umbral, lo que implica nueve veces menos sumadores, cero XNOR y la mitad del coste *hardware* para detectar el pico de la correlación. El pequeño número y tamaño de los sumadores del este TSA hacen que la latencia del sistema sea más baja que la de otros algoritmos.

3.2.1.5 Verificación Experimental

El algoritmo propuesto en esta sección ha sido evaluado utilizando el banco de pruebas descrito en el capítulo 2 y contrastado con el comportamiento de otros dos TSAs ([92] y [98]). Park en 2003 propuso un algoritmo de sincronización para canales inalámbricos el cual se basa en la autocorrelación de la señal recibida [92]. El segundo algoritmo de esta comparativa fue propuesto en 2014 para sistemas OFDM ópticos y se basa en el uso de la correlación cruzada. Para evaluar las prestaciones de estos algoritmos se utilizaron dos tipos de montaje, uno OBTB y otro con 100 km de fibra monomodo. Se obtuvo que para una BER de 10^{-2} , en ambos esquemas, los 3 algoritmos detectaron todos los paquetes sin errores (PCTD=1).

La Figura 3.17 muestra la probabilidad de obtener una detección de tiempo correcta frente a la potencia óptica recibida (*Received Optical Power* - ROP) para una configuración OBTB y de la transmisión a través de 100 km SSMF. Para conocer la probabilidad de detectar correctamente sincronismo con una BER de 10^{-2} , el peor de los umbrales para el FEC, es necesario saber el valor de ROP que satisface dicho umbral. Una vez obtenido este valor de potencia se puede ingresar al eje de abscisas de la Figura 3.17 y obtener el valor de PCTD.

Se ha realizado un estudio de la variación de la BER en función de la ROP, para lo cual se han transmitido 1.200 símbolos OFDM de prueba para cada algoritmo y para cada potencia recibida y, posteriormente, en el ordenador se han obtenido las tasas de errores. La Figura 3.18 muestra las curvas de BER frente a las variaciones de la ROP para una configuración OBTB y de 100 km SSMF; en ambas curvas se ha resaltado la ROP necesaria para obtener una BER de 10^{-2} . Los resultados de BER se calcularon sólo para las tramas detectadas correctamente; en este caso, todos los TSAs dan los mismos resultados y la figura caracteriza el comportamiento del sistema completo después de la sincronización.

Se observa en la Figura 3.17 que los tres algoritmos pueden sincronizar en tiempo con precisión para el valor BER de referencia, aunque el método de Park tiene un rendimiento menor para valores inferiores de ROP. Los métodos

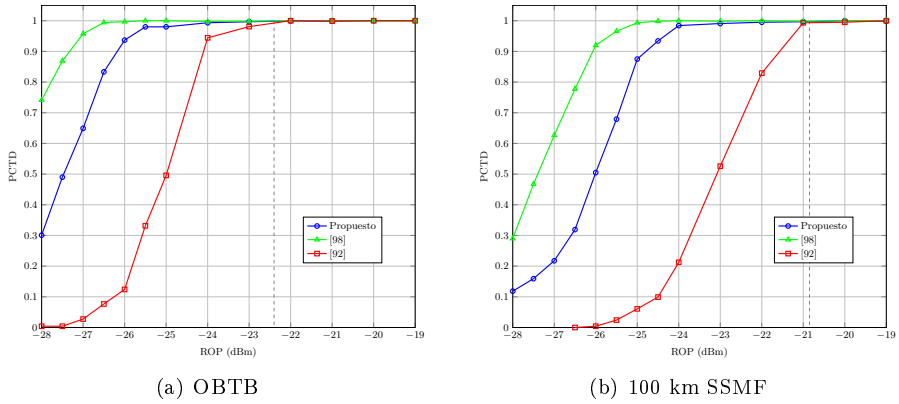


Figura 3.17: Curvas de PCTD versus ROP para diferentes esquemas de medición y el marcador para una BER de 10^{-2} .

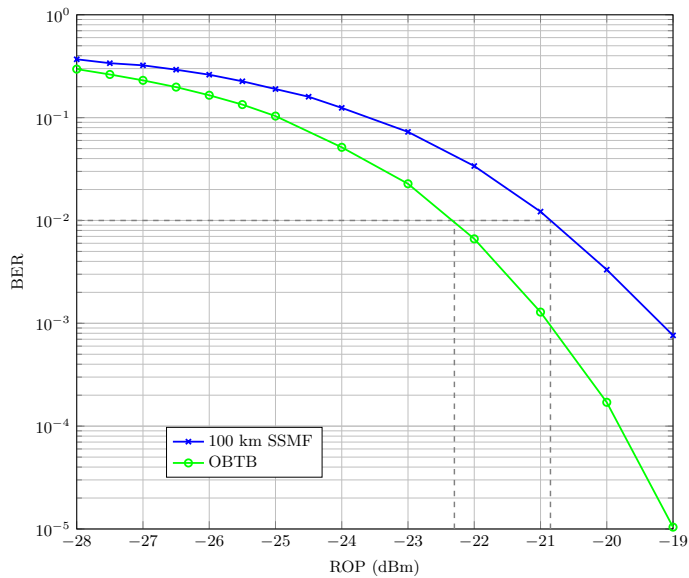


Figura 3.18: Curvas de BER versus ROP para diferentes esquemas de medición y el marcador para una BER de 10^{-2} .

que se basan en la correlación cruzada tienen un mejor rendimiento y este aumenta con el tamaño de la correlación. El TSA que se ha propuesto en [98] continúa detectando bien incluso para valores de potencia menores al umbral

gracias a que utiliza un tamaño de correlación mayor al utilizado en esta tesis, lo cual implica un mayor coste computacional. Sin embargo, la diferencia entre ambos algoritmos en el experimento de 100 km aparece solo para niveles de ROP por debajo de -23 dBm, correspondientes a valores de BER inferiores a 10^{-1} , es decir, correspondientes a niveles de potencia óptica que no son útiles en un escenario real. El mismo comportamiento se puede notar en el caso OBTB. Para resumir, los tres algoritmos presentan una PCTD de 1 para umbrales de BER correspondientes a SD-FEC y HD-FEC tanto en el caso de OBTB como de 100 km SSMF, razón por la cual lo que se prima a la hora de la elección del algoritmo es el coste computacional analizado en el apartado anterior.

3.2.2 Transformada Rápida de Fourier (FFT)

La transformada rápida de Fourier y su inversa [102] son algunos de los algoritmos más exigentes presentes en un sistema OFDM de muy alta velocidad debido a su alta carga computacional. En 1969 Bergland hizo una clasificación de las arquitecturas de FFT desde el punto de vista de implementación *hardware* [103]. El autor las ha dividido en cuatro grandes grupos de implementación: secuencial (basada en memorias), en cascada (*pipelined*), paralela y matricial (*fully parallel*). Cada uno de ellas supone una relación de compromiso diferente entre el *throughput* y la complejidad (área). Para maximizar el *throughput* se deben utilizar arquitecturas *hardware* totalmente paralelas. En implementaciones matriciales (p. ej. [104-106]) cada suma/multiplicación del gráfico de flujo del algoritmo de FFT se traduce en un sumador/multiplicador en el *hardware*. Cuando la FFT es de muchos puntos se utilizan demasiados recursos y esta arquitectura se vuelve casi inviable para dispositivos FPGA. Las arquitecturas tipo secuencial (p. ej. [107, 108]) son adecuadas para minimizar el área. Este tipo de implementación *hardware* calcula la FFT de forma iterativa utilizando datos almacenados en memoria, su bajo *throughput* hace que no sea adecuada para sistemas de alta velocidad. Las arquitecturas paralelas procesan N muestras en simultáneo durante $\log_R(N)$ ciclos de reloj, donde N es la longitud de la FFT y R la base (*radix*) del algoritmo, pero debido a las dependencias existentes entre las sucesivas etapas del algoritmo no se puede utilizar segmentación y esto limita el *throughput* de la implementación. Por último, las arquitecturas *pipelined* FFT (p. ej. [109]) funcionan con R entradas en paralelo, con $\log_R(N)$ etapas en cascada y un elemento computacional (CE) por etapa. La arquitectura FFT segmentada (*Pipelined FFT* - PFFT) es la opción elegida para aplicaciones de alta velocidad, ya que es muy regular, se puede escalar, paralelizar y segmentar.

Para aumentar la velocidad de procesamiento de las arquitecturas PFFT, mientras se evita el considerable aumento en *hardware* de una implementación totalmente paralela, se introduce un cierto grado de paralelismo a la implementación dando como resultado lo que se denomina arquitectura PFFT paralela (*Parallel PFFT* - PPFFT) [110]. Este tipo de diseño *hardware* aumenta la cantidad de CE en cada etapa para obtener el aumento de *throughput* requerido, manteniendo un 100 % de eficiencia de utilización del *hardware* (*Hardware Utilization Efficiency* - HUE). Este tipo de arquitecturas (p. ej. [111-115]) se clasifican en las que utilizan retardos conmutados multicamino (*Multi-path Delay Commutator* - MDC) o retardos realimentados simples (*Single-path Delay Feedback* - SDF), para reorganizar los datos que ingresan a los diferentes CE; MDC ha demostrado ser el enfoque más eficiente para implementaciones de alto rendimiento [112]. Este tipo de arquitectura ha sido el elegido para implementar la FFT requerida en nuestro sistema.

A continuación se presenta el diseño de una arquitectura PPFFT con decimación en frecuencia (*Decimation-In-Frequency* - DIF), en base 4 (*radix-4*), utilizando MDC, con 4 caminos de datos (O) y procesando 16 muestras en paralelo (P) basado en la arquitectura descrita en [110]. Después, este diseño se modifica para realizar el computo de una FFT de longitud variable de 16, 64, 256 y 1024 puntos, lo cual permitirá en un futuro realizar de manera sencilla estudios sobre los efectos de la longitud de la FFT en los sistemas OFDM de comunicaciones ópticas. También se mostrará el diseño de un módulo de reordenamiento de muestras de entrada y salida de la FFT para que las mismas ingresen y egresen en orden natural independientemente de la longitud de FFT utilizada.

Aunque en esta tesis no se ha realizado el diseño e implementación del módulo IFFT que integra cualquier transmisor OFDM, vale la pena mencionar que tan solo se requieren cambios menores para implementar las transformaciones inversa a partir de la versión directa.

3.2.2.1 PPFFT de longitud fija

En [110] se describen arquitecturas PPFFT con decimación en el tiempo, de base 2 (*radix-2*) y que utilizan MDC para diferentes caminos de datos (uno, dos y cuatro), y luego se extiende a *radix-4* para un caso específico. En la Figura 3.19 se muestra el grafo de la arquitectura generalizada *radix-4* MDC PPFFT con $O=4$ ($P = 16$) desarrollada en esta tesis, y basada en las ideas y pautas descritas en [110]. Una característica importante de esta arquitectura es que solo utiliza dos tipos de bloques: elementos computacionales (*Computational*

Elements - CEs) y conmutadores de demoras (*Delay Commutators* - DCs). Cada CE está compuesto por una unidad de computación atómica que en la literatura se denomina mariposa o *butterfly* [102]. Mientras que cada DC cumple la función de reordenar las muestras entre los CEs de las diferentes etapas.

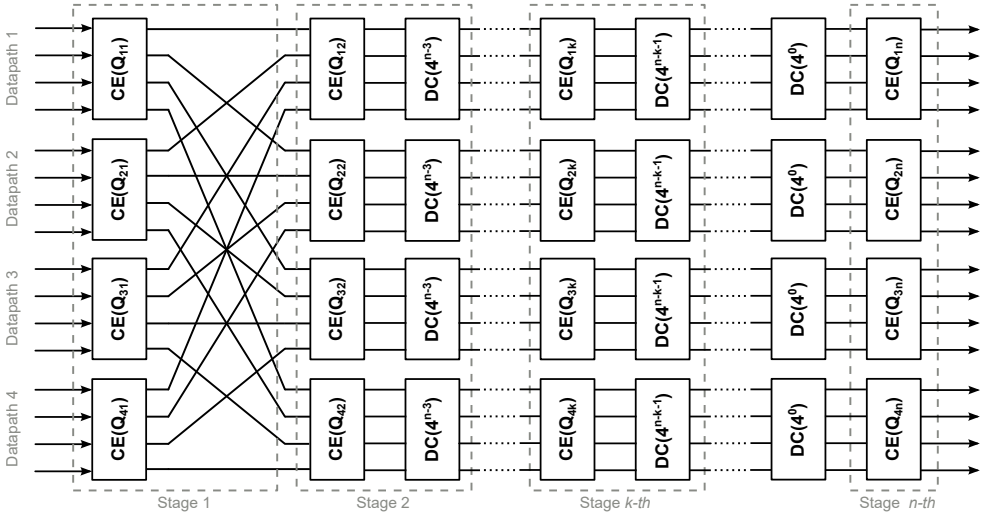


Figura 3.19: Arquitectura MDC PPFFT con $O=4$ y $R=4$ de n etapas.

Por regla general, el algoritmo de FFT está compuesto de n etapas, donde $n=\log_R(N)$, y a cada etapa se la numera con el índice k , donde k puede tomar valores entre 1 y n . En cada etapa se deben ejecutar N/R mariposas con diferentes factores de rotación (*Twiddle Factors* - TFs). En las arquitecturas PPFFT existen O (P/R) elementos computacionales por etapa y cada uno debe realizar $N/R/O$ mariposas. Para el caso de $O=4$ y $R=4$, se necesitan 4 CEs por etapa y cada CE procesa $N/16$ mariposas con diferentes conjuntos de factores de rotación. La velocidad de conmutación y el número de elementos de retardo de los DCs depende de las etapas que estos interconecten.

La arquitectura de la Figura 3.19 difiere de una FFT segmentada de *radix*-4 ($R=4$ y $O=1$) en que permite procesar 16 muestras simultáneas ($P=16$, $R=4$ y $O=4$), cuadruplicando el *throughput*, manteniendo la HUE en 100 % y cuadruplicando la cantidad de CEs por etapa. Por el contrario, la cantidad de DCs no se cuadruplica porque los DCs de la primera etapa desaparecen y son reemplazados por simples conexiones entre las salidas de los CEs de la primera etapa ($k=1$) y las entradas de los CEs de la segunda etapa ($k=2$). Como se

verá más adelante, los DCs de la primera etapa en una arquitectura PFFT son los que más latencia introducen y más recursos *hardware* utilizan.

En el algoritmo FFT con decimación en frecuencia las mariposas de la primera etapa procesan R muestras de entrada que están equiespaciadas en el tiempo N/R , lo cual implica que las muestras no ingresan en orden natural en una arquitectura PFFT. A la salida del algoritmo FFT las muestras se presentan en un orden denominado *digit-reversal*, lo cual implica que las muestras no egresan en orden natural en una arquitectura PFFT. Por lo tanto, tanto las muestras de entrada como las de salida deben ordenarse de una manera especial y ésta dependerá de la longitud de la FFT (N), de la base utilizada (R) y del número de caminos de datos (O) [116]. En el apartado 3.2.2.3 se darán todos los detalles acerca de este reordenamiento.

En Figura 3.19 se puede ver que, después de los CEs de la segunda etapa, las 4 rutas de datos no se vuelven a cruzar. Esto sucede independientemente de la longitud de la FFT y, a excepción de la primera etapa, los CEs y DCs de las 4 rutas de datos son iguales. Para cada etapa, los factores de rotación son una versión diezmada de la anterior y la última etapa tiene todos sus TFs iguales a la unidad. Todas estas características permiten diseñar un procesador FFT de longitud configurable con una lógica adicional mínima, tal como se verá en el apartado 3.2.2.2.

3.2.2.1.1. Elemento Computacional

El elemento computacional es el bloque que realiza las operaciones matemáticas con números complejos del algoritmo de la FFT. Comúnmente a este bloque se le denomina mariposa (*butterfly*) y el número de entradas y salidas, como así también la cantidad de operaciones complejas, dependen únicamente de la base del algoritmo FFT (R). Un elemento computacional de un algoritmo FFT de *radix-4* se corresponde con una DFT de 4 puntos cuyas salidas se multiplican por 4 números complejos (factor de rotación). De esta manera, se puede formular el comportamiento de un CE de *radix-4* mediante la ecuación 3.12, donde las cuatro entradas complejas se denominan A, B, C y D, y donde las cuatro salidas complejas se denominan E, F, G, y H.

$$\begin{aligned}
 E &= (A + B + C + D) \cdot W_N^{0Q} \\
 F &= (A - jB - C + jD) \cdot W_N^{1Q} \\
 G &= (A - B + C - D) \cdot W_N^{2Q} \\
 H &= (A + jB - C - jD) \cdot W_N^{3Q}
 \end{aligned}
 \tag{3.12}$$

Esta ecuación tiene 6 variables: 4 son las entradas complejas al bloques, otra es Q y la otra es N . El factor de rotación se puede expresar como $W_N = e^{-j2\pi/N}$, un número complejo de módulo uno y fase variable. Q es un índice dependiente de la posición del algoritmo que define el valor de los 4 factores de rotación, donde el primer factor de rotación ($W_N^{0Q} = e^{-(j2Q\pi/N)\cdot 0}$) siempre vale 1 independientemente de los valores de Q y N . A esta mariposa *radix-4*, utilizada en el algoritmo de FFT, en ocasiones se la puede encontrar en la literatura técnica como *dragonfly* [117].

La *dragonfly* descrita en ecuación 3.12 requiere de 12 sumas de números complejos, 3 multiplicaciones de números complejos ($W_N^{0Q} = 1$) y 8 multiplicaciones triviales ($j, -j, -1$). Si se opera algebraicamente la ecuación 3.12, las 12 sumas de números complejos (24 sumadores reales) y las 8 multiplicaciones triviales (lado izquierdo de la ecuación 3.12) se pueden reescribir como 16 sumas reales, tal como se muestra en la ecuación 3.13. En colores se han resaltado los 4 grupos de operaciones que operan sobre 2 variables de entrada, los cuales se pueden implementar mediante 4 sumadores complejos (primera etapa). Después, estos resultados parciales se suman o restan para completar las operaciones de la ecuación 3.13 (segunda etapa), lo cual requiere de 4 sumadores complejos. Resumiendo, la ecuación 3.13 requiere de 8 sumadores complejos (16 sumadores reales) agrupados en dos etapas (lo cual es una ventaja a la hora de la implementación), esto implica una reducción del coste computacional de 8 sumadores reales respecto de la implementación directa de la ecuación 3.12.

$$\begin{aligned}
 \text{Re}(A + B + C + D) &= [\text{Re}(A) + \text{Re}(C)] + [\text{Re}(B) + \text{Re}(D)] \\
 \text{Im}(A + B + C + D) &= [\text{Im}(A) + \text{Im}(C)] + [\text{Im}(B) + \text{Im}(D)] \\
 \\
 \text{Re}(A - jB - C + jD) &= [\text{Re}(A) - \text{Re}(C)] - [\text{Im}(D) - \text{Im}(B)] \\
 \text{Im}(A - jB - C + jD) &= [\text{Im}(A) - \text{Im}(C)] - [\text{Re}(B) - \text{Re}(D)] \\
 \\
 \text{Re}(A - B + C - D) &= [\text{Re}(A) + \text{Re}(C)] - [\text{Re}(B) + \text{Re}(D)] \\
 \text{Im}(A - B + C - D) &= [\text{Im}(A) + \text{Im}(C)] - [\text{Im}(B) + \text{Im}(D)] \\
 \\
 \text{Re}(A + jB - C - jD) &= [\text{Re}(A) - \text{Re}(C)] + [\text{Im}(D) - \text{Im}(B)] \\
 \text{Im}(A + jB - C - jD) &= [\text{Im}(A) - \text{Im}(C)] + [\text{Re}(B) - \text{Re}(D)]
 \end{aligned}
 \tag{3.13}$$

Por otro lado, cada multiplicación de números complejos de la *dragonfly* (lado derecho de la ecuación 3.12) se puede llevar a cabo de forma directa utilizando 4 multiplicadores reales y 2 sumadores reales, tal como se muestra en la ecuación 3.14.

$$\begin{aligned} Re(X \cdot Y) &= [Re(X) \cdot Re(Y)] - [Im(X) \cdot Im(Y)] \\ Im(X \cdot Y) &= [Im(X) \cdot Re(Y)] + [Re(X) \cdot Im(Y)] \end{aligned} \quad (3.14)$$

Existe la posibilidad de reducir el coste computacional de esta operación a 3 multiplicaciones y 5 sumas entre números reales, tal como se muestra en la ecuación 3.15. Pero como se detallará en el apartado 4.2.3.1, esta versión de multiplicador complejo no es más eficiente que la implementación directa de la ecuación 3.14 utilizando los recursos disponibles en la FPGA.

$$\begin{aligned} Re(X \cdot Y) &= Re(X) \cdot [Re(Y) + Im(Y)] - Im(Y) \cdot [Re(X) + Im(X)] \\ Im(X \cdot Y) &= Re(X) \cdot [Re(Y) + Im(Y)] + Re(Y) \cdot [Im(X) - Re(X)] \end{aligned} \quad (3.15)$$

En la Figura 3.20 se muestra el esquema del elemento computacional basado en las ecuaciones 3.13 y 3.14. En ella se puede ver claramente dos 2 etapas de 4 sumadores complejos cada una y una tercer etapa con 3 multiplicadores complejos. Por lo tanto, el coste computacional de la *dragonfly* de la Figura 3.20 es de 22 sumadores reales y 12 multiplicadores, lo cual implica un ahorro de 8 sumadores reales con respecto a la implementación directa de la ecuación 3.12.

En la etapa k -ésima del algoritmo de FFT, con $k=1, 2, \dots, \log_4(N)$, se procesan $N/4$ mariposas que pueden ser agrupadas en 4^{k-1} grupos, es decir, se computan $(N/4)/4^{k-1}$ mariposas por grupo. Estos grupos tienen la particularidad de que sus TFs de sus CEs son iguales dentro de una misma etapa. Los valores de Q por grupo para diferentes longitudes de FFT se pueden expresar en pasos de 4^{k-1} desde 0 hasta $N/4 - 1$, y estos valores se resumen en Tabla 3.2. Para $N=1024$, los valores de Q de la etapa 2 (4 grupos de 64 mariposas) son 0:4:255 (primero:incremento:último) y para $N=256$ los valores de Q de la etapa 1 (1 grupo de 64 mariposas) son 0:1:63. Aunque a priori parezca que estos dos vectores no guardan relación algunas si se analizan los valores de los TFs (W_N^{1Q} , W_N^{2Q} y W_N^{3Q}) para estos dos vectores de valores de Q se observan que son equivalentes. Es decir, los TFs de la segunda etapa ($k=2$) de una FFT de 1024 puntos son los mismos que los de la primera etapa ($k=1$) de una FFT de 256 puntos. Esta propiedad se puede hacer extensiva a las restantes etapas y diferentes longitudes; en tres colores diferentes se resaltan en la Tabla 3.2

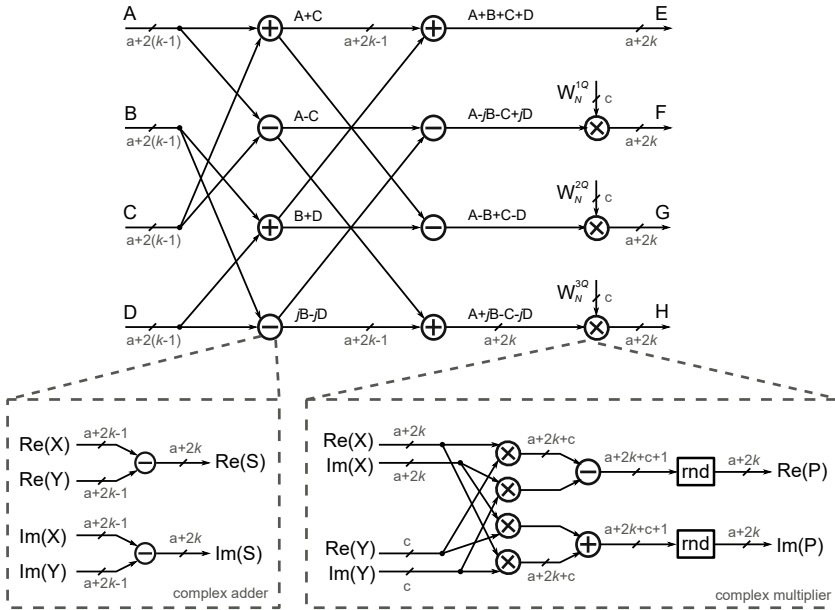


Figura 3.20: Elemento computacional *radix-4* con decimación en frecuencia.

los TFs equivalentes. De este mismo análisis se desprende que los TFs de la primera etapa de una FFT de 256 puntos son los mismos que los de la primera etapa de una FFT de 1024 puntos diezmos por 4, esta propiedad se hace extensiva para las diferentes valores N con distinto factor de diezmo. Estas características de los TFs facilitan la tarea de realizar un algoritmo FFT de longitud variable, tal como se verá en el apartado 3.2.2.2.

Tabla 3.2: Valores de Q de las diferentes etapas de una FFT de N puntos

$N \backslash k$	1	2	3	4	5
1024	0:1:255	0:4:255	0:16:255	0:64:255	0
256	0:1:63	0:4:63	0:16:63	0	-
64	0:1:15	0:4:15	0	-	-
16	0:1:3	0	-	-	-

Volviendo al caso de la arquitectura PPFFFT con 4 caminos de datos, cada elemento computacional de la Figura 3.19 calcula $N/16$ mariposas en cada una de las etapas y como se mencionó al comienzo del párrafo anterior, estas pueden

ser agrupadas $(N/4)/4^{k-1}$ mariposas por grupo. En la primera etapa, hay un solo grupo de $N/4$ mariposas y $N/4$ valores diferentes de Q que se reparten en partes iguales (Q_{11} , Q_{21} , Q_{31} y Q_{41}) entre los 4 CEs. En la segunda etapa, hay 4 grupos de $N/16$ mariposas y $N/16$ valores diferentes de Q , por lo tanto, los valores de Q para los 4 CEs son los mismos ($Q_{12} = Q_{22} = Q_{32} = Q_{42}$). Esto mismo sucede para las restantes etapas.

Finalmente, el coste computacional total de la arquitectura PPFFFT *radix*-4 de N puntos y P muestras en paralelo se puede expresar mediante la ecuación 3.16.

$$\begin{aligned} & 8 \cdot P/4 \cdot \log_4(N) && \text{sumadores complejos} \\ & 3 \cdot P/4 \cdot [\log_4(N) - 1] && \text{multiplicadores complejos} \end{aligned} \quad (3.16)$$

3.2.2.1.2. Conmutador de demoras

El conmutador de demoras reordena los datos complejos que egresan del CE de una etapa para ingresar en el CE de la siguiente etapa en un orden específico. Este orden depende de los valores de N , P , R del algoritmo PPFFFT y de las etapas que esté vinculando. El DC para la arquitectura FFT segmentada de *radix*- R se describe en [118] y se muestra en la Figura 3.21.

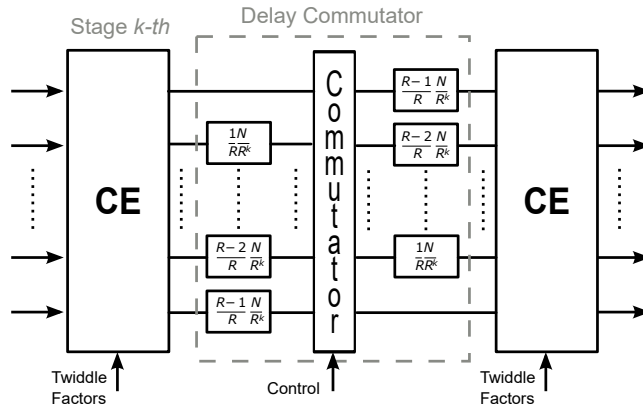


Figura 3.21: Conmutador de demoras de la etapa k -ésima de la FFT segmentada con *radix*- R .

Con $R=4$ y $N=4^n$, cuando los datos ingresan al DC pasan a través de 4 líneas de retardo paralelas. No se aplica ningún retardo a los datos de la primera fila, en la segunda fila los datos son retardados 4^{n-k-1} ciclos de reloj, en la tercera

fila los datos son retardados $2 \cdot 4^{n-k-1}$ ciclos de reloj y en la cuarta fila los datos son retardados $3 \cdot 4^{n-k-1}$ ciclos de reloj. Posteriormente, los datos pasan a través de un conmutador para intercambiar las filas y las columnas. Por último, los datos se retrasan nuevamente como se muestra en la Figura 3.21. Si los datos llegan al módulo cada T segundos, el conmutador debe modificar su selección cada $f \cdot T$ segundos; donde $f=4^{n-k-1}$. A medida que N aumenta, se incrementan los retardos introducidos en la ruta de datos dentro de los DCs de cada etapa, lo que implica un aumento en la latencia del sistema. La Tabla 3.3 muestra los valores de f para diferentes valores de N de una arquitectura FFT segmentada con $R=4$.

Tabla 3.3: Valores de f para diferentes longitudes de FFT con $R=4$

$N/n \backslash k$	1	2	3	4
1024/5	64	16	4	1
256/4	16	4	1	-
64/3	4	1	-	-
16/2	1	-	-	-

Para un algoritmo PFFFT de *radix-4* que procesa 16 muestras en paralelo ($P=16$, $R=4$ y $O=4$), los DCs de la primera etapa ($k=1$ y $f=64$) desaparecen y son reemplazados por conexiones directas entre los ECs de la primera y segunda etapa. Esto provoca que la latencia total de la arquitectura PFFFT (ver la Figura 3.19 y la Tabla 3.3) disminuya significativamente con respecto a la de la arquitectura PFFT, aproximadamente 4 veces. La latencia teórica de la arquitectura PFFFT de *radix-4* se puede expresar mediante la ecuación 3.17. Cabe señalar que esta latencia es independiente de la base utilizada y disminuye con el grado de paralelismo P .

$$\text{Lat} = \frac{N - P}{P} \tag{3.17}$$

El enrutamiento de datos que se produce en un DC de $R=4$ y con $f=4$ se representa gráficamente en la Figura 3.22. Los datos de entrada numerados del 0 al 63 se muestran en cuatro flujos paralelos a la izquierda de Figura 3.22. Transforma 4 flujos de datos separados por 16 puntos en 4 flujos donde los datos están separados por 4 puntos. Este proceso se deriva y explica con mayor detalle en [118].

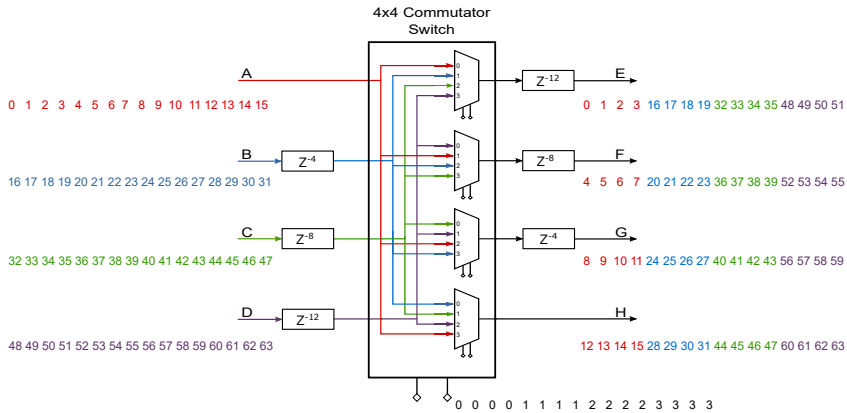


Figura 3.22: Patrones de datos a través del conmutador de demoras de $R=4$ con $f=4$.

3.2.2.2 PPFFT de longitud variable

A continuación se presenta el diseño de un procesador FFT de longitud variable, configurable en tiempo de ejecución, que puede realizar el cómputo de una FFT de 16, 64, 256 y 1024 puntos. El mismo se basa en el diseño de una PPFFT de *radix*-4 y 1024 puntos, con decimación en frecuencia y 16 entradas/salidas complejas en paralelo.

Analizando la estructura de la Figura 3.19 y los valores Q y f presentados en las Tablas 3.2 y 3.3, respectivamente, se puede concluir que una longitud menor a 1024 se puede calcular eliminando etapas de la arquitectura PPFFT de 1024 puntos y modificando algunos factores de rotación. Para eliminar las etapas que no se desean utilizar, se realiza una bifurcación en el camino de datos mediante multiplexores que se introducen en la arquitectura, de manera tal que los datos atraviesen los CEs y DCs que corresponden según la longitud de la FFT seleccionada. El diagrama de bloques del procesador de FFT de longitud variable propuesto basado en la arquitectura PPFFT con MDC y *radix*-4 se representa en la Figura 3.23, donde los bloques rectangulares representan los CEs y DCs de cada etapa (denominados S1 a S5). Tenga en cuenta que los bloques S_k se corresponden con las etapas de Figura 3.19.

Como se explica en el apartado 3.2.2.1, esta arquitectura tiene la peculiaridad de que los CDs de la primera etapa se reemplazan por conexiones directas entre las salidas de los CEs de la primera etapa y las entradas de los CEs de las segunda etapa, y estas conexiones son las mismas independientemente de cual sea la longitud de la FFT. Para aprovechar este hecho, se necesita un

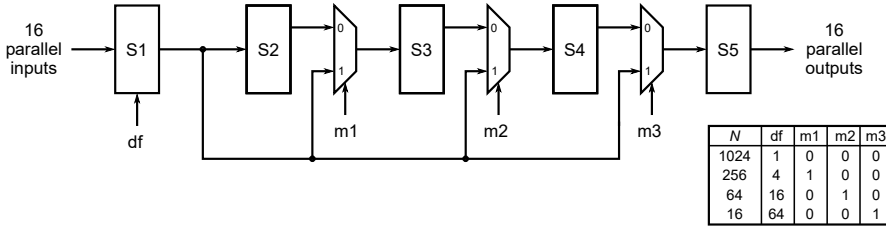


Figura 3.23: Arquitectura de longitud variable R4MDC PPFFT con 4 caminos de datos. S_k simboliza las etapas FFT, incluidos los bloques CEs y DCs.

método para generar los TFs de la primera etapa para diferentes longitudes de FFT a partir de los TFs de la FFT de 1024 puntos. Si se analiza la Tabla 3.2, se puede ver que los valores de Q en la etapa 1 para $N=256$ son lo mismo que $N=1024$ pero diezmados por 4. Para $N=64$ el factor de diezmado es 16 y para $N=16$ es 64. En general, $df=1024/N$, donde df es el factor de diezmado. Como ejemplo, para el caso de $N=64$, se debe crear una bifurcación entre las entradas del bloque S2 y S4 ($m=010$), y los factores de la primera etapa deben ser diezmados por 16. Es importante tener en cuenta que también cambia el orden de muestras que entran y salen del procesador FFT.

3.2.2.3 Reordenamiento de entradas y salidas

En la mayoría de los sistemas donde se utiliza un procesador FFT, y en particular en un sistema de comunicación OFDM, tanto las muestras de entrada como las de salida, requieren entrar y salir del módulo en un orden natural. Como se ha mencionado anteriormente, las entradas y salidas del algoritmo PPFFT no están organizadas en orden natural, y su orden depende de la longitud de la FFT y del número de muestras en paralelo (es decir, de N y P). Bajo estas circunstancias es necesario realizar un reordenamiento de las muestras antes y después de la FFT.

Las muestras de entrada y salida se pueden interpretar como una matriz de índices bidimensional con P filas y L columnas, donde $L=N/P$. La matriz de índices para las muestras en orden natural y de entrada para un algoritmo R4MDC PPFFT de N puntos y con 16 muestras paralelas, se presenta en Tabla 3.4 y Tabla 3.5 respectivamente, donde $M=N/4$, i es el índice del número de entrada (de 0 hasta $P-1$) y t es el número de ciclo de reloj (de 1 hasta N/P). La matriz de índices para las muestras de salida se obtiene aplicando la transformada *digit-reversal* de base 4 a la matriz de índices para las muestras de entrada.

Tabla 3.4: Orden natural para PPFFT con $P=16$

$i \backslash t$	1	2	$N/16$
0	0	16	$16(L-1)$
1	1	17	$16(L-1)+1$
\vdots	\vdots	\vdots	\vdots
15	15	31	$N-1$

Tabla 3.5: Orden de entrada para PPFFT con $P=16$

$i \backslash t$	1	2	$N/16$
0	0	1	$L-1$
1	M	$1+M$	$L-1+M$
2	$2M$	$1+2M$	$L-1+2M$
3	$3M$	$1+3M$	$L-1+3M$
4	L	$L+1$	$2L-1$
5	$L+M$	$L+1+M$	$2L-1+M$
6	$L+2M$	$L+1+2M$	$2L-1+2M$
7	$L+3M$	$L+1+3M$	$2L-1+3M$
8	$2L$	$2L+1$	$3L-1$
9	$2L+M$	$2L+1+M$	$3L-1+M$
10	$2L+2M$	$2L+1+2M$	$3L-1+2M$
11	$2L+3M$	$2L+1+3M$	$3L-1+3M$
12	$3L$	$3L+1$	$4L-1$
13	$3L+M$	$3L+1+M$	$4L-1+M$
14	$3L+2M$	$3L+1+2M$	$4L-1+2M$
15	$3L+3M$	$3L+1+3M$	$4L-1+3M$

Es importante tener en cuenta que para obtener la matriz de índices de las muestras de entrada al procesador FFT a partir de una matriz de índices en orden natural, es necesario realizar permutaciones en ambas dimensiones: espacial (filas) y temporal (columnas). Lo mismo sucede para obtener una matriz de índices en orden natural a partir de la matriz de índices de las muestras de salida del procesador FFT. A modo de ejemplo, la Figura 3.24 muestra el orden de los datos en la entrada y la salida para una PPFFT de 64 puntos. Se observa que la reorganización aplicada a las muestras de entrada es diferente de la aplicada a las muestras de salida y que se han realizado permutaciones ambas dimensiones.

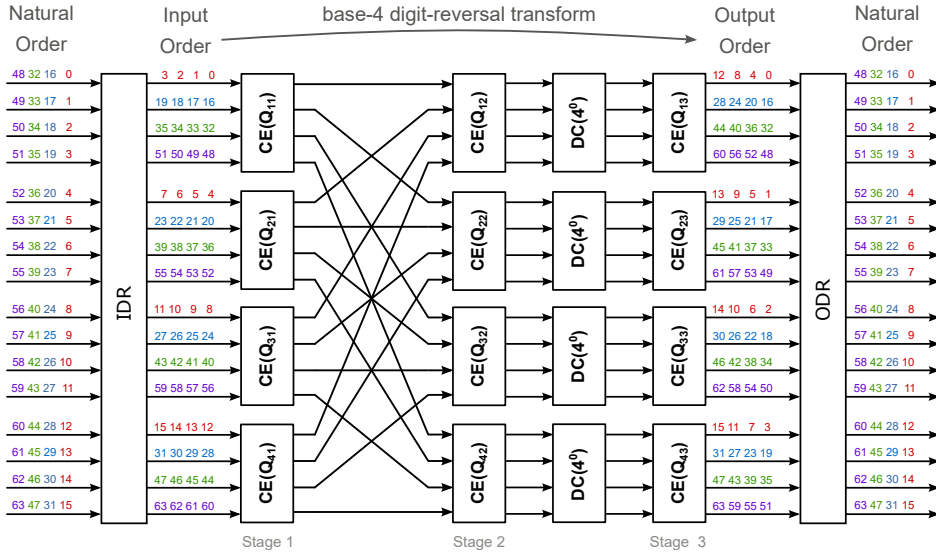


Figura 3.24: Patrones de datos de entrada/salida a través de la arquitectura R4MDC PPFFT 64 puntos con 4 caminos de datos ($N=64$ y $P=16$).

3.2.2.4 Modelo de precisión finita

Desde los años 70 ([119-122]) hasta la actualidad ([123-125]), se han realizado estudios acerca de los errores de precisión finita en los algoritmos FFT y todos ellos coinciden en analizar tres fuentes de error: redondeo (*roundoff*), desbordamiento (*overflow*) y cuantificación de coeficientes. En el caso de la FFT los errores de redondeo se producen cuando los resultados de las multiplicaciones complejas de cada CE son truncados o redondeados al tamaño de palabra del sistema. Los errores de desbordamiento se producen cuando las salidas del CE exceden el máximo tamaño de palabra permitido. Por último, los errores de cuantificación de coeficientes son el resultado de la representación de los factores de rotación con un número limitado de bits.

Este algoritmo tiene la particularidad de que si su entrada ($x_{[n]}$) está acotada entre ± 1 , su salida ($X_{[n]}$) estará acotada entre $\pm N$. Esto implica un crecimiento implícito de $\log_2(N)$ bits en la ruta de datos, porque de no ser así, podría haber desbordamientos. Existen distintas técnicas que evitan el crecimiento de la ruta de datos ($|X_{[n]}| < 1$) y los desbordamientos, algunas introducen un coste computacional adicional y otras no hacen un uso eficiente del rango dinámico. Las tres más utilizadas y ordenadas por el uso eficiente del rango dinámico son: escalado de la señal de entrada por $1/N$, escalado distribuido a

la entrada de cada etapa (p. ej. para *radix-2* este valor aproximadamente 0,5) y escalado dinámico o más conocido como *block floating point*. Cada una de las técnicas de escalado y la opción de dejar crecer el camino de datos, afectan de diversa manera al análisis de errores de redondeo como se verá más adelante.

Desde el punto de vista de la implementación *hardware*, los multiplicadores complejos (ecuación 3.12) de los diferentes CEs son el punto más conflictivo del diseño y desde el punto de vista del análisis de precisión finita, son los que producen los errores de redondeo. Según el análisis realizado en [117] acerca de cómo este tipo de errores afecta a la SNR a la salida de la FFT teniendo en cuenta la técnica utilizada para evitar desbordamientos, se obtuvo que: si se utiliza escalado de entrada ($x_{[n]}/N$) la $SNR = 2^{2b}/N^2$, si se utiliza escalado distribuido la $SNR = 2^{2b}/2N$ y si simplemente se deja crecer la ruta de datos la $SNR = 2^{2b}$; donde b es la cantidad de bits con la que se redondea la salida de cada multiplicador. Este estudio viene a decir que cuando no se utiliza ninguna estrategia de escalado, la SNR obtenida a la salida de la FFT es independiente de la longitud de la misma y es significativamente mejor que la obtenida por las otras dos técnicas de escalado. Este análisis ha sido realizado para algoritmos FFT base 2 con decimación en tiempo, sin contemplar el ruido de cuantificación de la señal de entrada y asumiendo que la misma es ruido blanco, considerando que la salida de todos los multiplicadores se redondea con la misma cantidad de bits, sin tener en cuenta el efecto de la cuantificación de los factores de rotación, ni que existen en los CE multiplicaciones triviales por ± 1 .

Extrapolando los resultados de SNR obtenidos en [117] a la arquitectura utilizada en este trabajo, se ha decidido dejar crecer la magnitud de la señal y no realizar ningún tipo de escalado. Es decir, si se utilizan a bits para cuantificar la señal de entrada (parte real e imaginaria), se obtienen $a + \log_2(N)$ bits a la salida de la FFT. Por otra parte, para garantizar una frecuencia de operación elevada se ha decidido utilizar los sumadores y multiplicadores *hardware* (DSP48) disponibles en la FPGA para implementar los multiplicadores complejos. Estas dos decisiones imponen ciertas restricciones al diseño que a continuación se detallan. Los multiplicadores *hardware* disponibles en la FPGA utilizada son de 25x18 bits. Considerando que 18 bits son más que suficientes para cuantificar los factores de rotación, se han escogido 25 bits para el camino de datos y 18 bits para los coeficientes. Esto implica que la salida de datos de los CEs de la penúltima etapa ($k = \log_4(N) - 1$) puede alcanzar como máximo un ancho de palabra de 25 bits, para la parte real y para la parte imaginaria, ya que en la última etapa los factores de rotación son unitarios (no se utilizan multiplicadores). Teniendo en cuenta que el crecimiento del camino de datos por etapa es de 2 bits para la parte real y 2 bits para la parte imaginaria,

se puede alcanzar como máximo 27 bits de ancho de palabra para los datos de salida de la FFT, lo cual restringe el ancho de palabra para las entradas ($27 - \log_2(N)$) a 23, 21, 19 y 17 bits para los diferentes longitudes de FFT de 16, 64, 256 y 1024 puntos, respectivamente.

En la Figura 3.20 se muestra el diagrama de flujo de la ecuación 3.12 y se detalla el esquema de precisión finita, donde a representa el ancho de palabra para las entradas de la FFT, c la cantidad de bits con los que se han cuantificado los factores de rotación y k es el número de etapa. Como no se utiliza ninguna técnica de escalado, después de las dos etapas de sumadores complejos, el tamaño del camino de datos aumenta 2 bits ($a + 2k$). A continuación se encuentran los multiplicadores complejos que sólo afectan a la fase de los datos internos del CE, con lo cual el tamaño de palabra de entrada y salida es el mismo ($a + 2k$). Dentro del multiplicador complejo (ver detalle de la Figura 3.20) el tamaño de palabra crece conforme a las operaciones que se van realizando, y es en la salida de datos donde se introduce el error por redondeo. Esto último es un caso más ventajoso que el analizado en [117], ya que allí se aplica el redondeo a la salida de cada multiplicador.

Existen muchos trabajos publicados acerca de los efectos de la precisión finita en el algoritmo de FFT, pero todos ellos son dependientes de múltiples factores: base, longitud, tipo de diezmado, tipo de escalado, tamaño de los multiplicadores, método de redondeo y tipo de señal de entrada, entre otros. Esto hace que sea obligatorio realizar una simulación del algoritmo de precisión finita diseñado, utilizando como entrada señales de OFDM, de manera tal de ver cómo afectan las decisiones adoptadas a las prestaciones del sistema de comunicaciones. En el marco en el cual se desarrolla esta tesis, la longitud más grande de FFT es de 1024 puntos, el orden de modulación transmitido más elevado es 512-QAM y el tamaño de palabra de entrada es de 10 bits (ADC). En la 3.25 se muestran las curvas de BER en función del SNR para un canal AWGN utilizando el algoritmo desarrollado en esta sección en 3 versiones diferentes: una de punto flotante y dos de punto fijo (redondeo y truncamiento). Se observa que las curvas de los 3 algoritmos son prácticamente iguales a la teórica para los valores de SNR que satisfacen una transmisión libre de errores utilizando SD-FEC o HD-FEC.

De esta manera queda demostrado que los tamaños de palabra utilizados y la no utilización de una técnica de escalado son adecuados para satisfacer las necesidades de este trabajo. Utilizar o no una técnica de redondeo en lugar de truncamiento, en este caso, es una decisión que depende del coste de implementación que se analizará en el apartado 4.2.3.

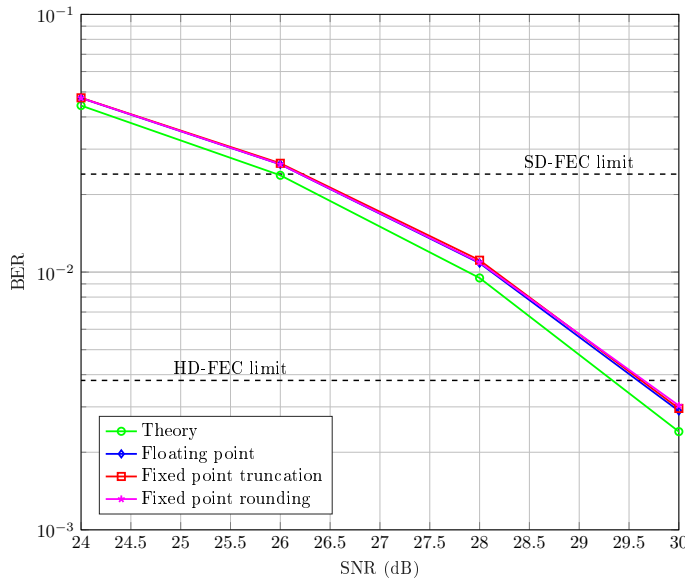


Figura 3.25: Curva de BER vs SNR para señal OFDM modulada con 512-QAM utilizando una FFT de 1024 puntos con 10 bits de entrada, 20 bits de salida y 18 bits de TF.

3.2.3 Ecualizador de canal

El objetivo de la ecualización de canal es reducir las distorsiones de amplitud y fase debido al canal y los elementos que constituyen el *front-end* analógico. Una de las principales ventajas de los sistemas OFDM es que permiten realizar una ecualización de baja complejidad en el dominio de frecuencia, que es mucho menos complejo que la ecualización en dominio del tiempo requerida en las modulaciones monoportadora. De hecho, las modulaciones multiportadora descomponen el canal selectivo en frecuencia en varios canales planos, permitiendo una ecualización de baja complejidad por tono que consiste en aplicar un ecualizador en dominio de la frecuencia (*Frequency-domain Equalizer* - FEQ) de una etapa por tono para compensar la amplitud del canal y los efectos de fase.

Una forma simple y directa de implementar un FEQ es dividir los símbolos recibidos por la respuesta del canal, ambos en el dominio de la frecuencia; esto se conoce como ecualización de forzado a cero (*Zero-Forcing Equalization* - ZFE). Con este fin, primero se realiza una estimación del canal en frecuencia y después se calculan los coeficientes del compensador de canal. Para estimar

la respuesta del canal se pueden insertar símbolos conocidos en el preámbulo o portadoras piloto dentro de los símbolos OFDM.

Estos símbolos que se transmiten en el preámbulo se los denomina símbolos piloto. La calidad de la estimación del canal depende del número y la disposición de los símbolos piloto. En el caso de portadoras piloto, para obtener la respuesta del canal es necesario algún tipo de interpolación: lineal, de segundo orden y cubic spline entre otras [126]. En ambos casos, cuanto mayor es el número de símbolos/portadoras piloto, más precisa es la estimación. Sin embargo, el número de símbolos/portadoras piloto afecta a la tasa de transmisión del sistema, por lo tanto, existe una relación de compromiso entre la precisión de la estimación del canal y la degradación de la tasa de transmisión.

Por otro lado, tres tipos de disposición de portadoras piloto son posibles, ya sea que el canal se desvanezca lentamente o sea variable en el tiempo: *comb-type*, *block-type* y *incline-type*. De hecho, en el caso de los canales de desvanecimiento lento, donde las características del canal no cambian rápidamente, la estimación del canal se realiza comúnmente insertando símbolos piloto en todas las subportadoras al comienzo de la trama OFDM. Esto se conoce comúnmente como disposición *block-type* [127]. Los canales ópticos alámbricos son canales casi invariables en el tiempo, razón por la cual, la disposición *block-type* es la indicada para una ecualización óptima tal como se demuestra en [127]. Una ventaja de esta disposición es que sólo se usan unos pocos símbolos OFDM para realizar la estimación del canal. Los símbolos OFDM restantes se usan para transportar datos útiles y no se desperdician más subportadoras para estimar el canal. Una vez que se estima la respuesta del canal, los coeficientes del ecualizador se calculan y se aplican a los símbolos recibidos.

3.2.3.1 Estimador de canal

Como se comentó en el apartado 3.2.1, los símbolos largos del preámbulo, cuyo comienzo ha sido determinado por el algoritmo de sincronización de tiempo, se usan para estimar la respuesta en frecuencia del canal. El algoritmo de estimación de canal utilizado en el presente trabajo se realiza en el dominio de la frecuencia. La secuencia de estimación está compuesta por 4 LS idénticos y las subportadoras desde la 1 hasta $N/2 - 1$ de cada LS se modulan utilizando desplazamiento de fase binaria (*Binary Phase Shift Keying* - BPSK). Como la señal generada debe tener un valor real, es necesario que las subportadoras desde -1 hasta $-N/2 + 1$ tengan simetría hermitica alrededor de la subportadora de corriente continua. Los contenidos de los cuatro símbolos largos son idénticos, por lo que se pueden promediar en el tiempo para mejorar la calidad

de la estimación del canal [128]. Este promedio puede ser calculado antes de la FFT gracias a que la FFT es una operación lineal, tal como se muestra en la ecuación 3.18,

$$LS_{rm}[k] = \sum_{n=0}^{N-1} \left(\frac{\sum_{m=0}^3 x[n + n_{gi} + mN_{ls}]}{4} \right) e^{-j2\pi kn/N}, \quad (3.18)$$

con $k = 0, 1, 2, \dots, N - 1$, donde $LS_{rm}[k]$ es la DFT del promedio de los LS recibidos y n_{gi} indica la muestra de la señal recibida en la cual comienza el primer LS.

La estimación en frecuencia del canal (\hat{H}) se obtiene de la siguiente manera:

$$\hat{H}[k] = \frac{LS_{rm}[k]}{LS_t[k]}, \quad (3.19)$$

donde $LS_t[k]$ se corresponde con la DFT del símbolo largo transmitido.

Para reducir el tamaño del preámbulo para el caso particular de $N=1024$, y de esta manera no penalizar tanto la tasa binaria final, se propone reducir el tamaño de los LS adoptando $N_{ls}=N/4$. En este caso la secuencia de estimación es generada modulando una de cada 4 subportadoras, mientras que las restantes subportadoras se rellenan con ceros. En el receptor primero se calcula la estimación del canal para las subportadoras activas y, posteriormente, se interpolan los valores de las restantes subportadoras tal como se indica en la ecuación 3.20. A diferencia de lo que sucede en los canales inalámbricos, la respuesta del canal óptico puede ser considerada prácticamente constante en un periodo corto de tiempo y las diferencias entre dos subportadoras contiguas muy pequeña. Es por esto que la respuesta del canal para las frecuencias de las subportadoras inactivas puede ser estimada mediante una interpolación lineal en frecuencia de las subportadoras activas, tal como se muestra a continuación.

$$\begin{aligned} \hat{H}_{i_i}[4k + 0] &= \hat{H}[k] \\ \hat{H}_{i_i}[4k + 1] &= \hat{H}[k] + 0,25 \cdot (\hat{H}[k + 4] - \hat{H}[k]) \\ \hat{H}_{i_i}[4k + 2] &= \hat{H}[k] + 0,50 \cdot (\hat{H}[k + 4] - \hat{H}[k]) \\ \hat{H}_{i_i}[4k + 3] &= \hat{H}[k] + 0,75 \cdot (\hat{H}[k + 4] - \hat{H}[k]) \end{aligned} \quad (3.20)$$

Donde $k = 1, 2, \dots, N/4 - 1$ y los valores de $\hat{H}_{li}[1]$, $\hat{H}_{li}[2]$ y $\hat{H}_{li}[3]$ no pueden ser interpolados. Razón por la cual no se podrán colocar datos útiles en estas 3 primeras subportadoras. A los valores obtenidos mediante la ecuación 3.20 se les debe aplicar un factor de escala de $1/\sqrt{4}$.

La paralelización del estimador de canal no presenta grandes dificultades, pero aun así, a continuación se analizarán las tres ecuaciones que intervienen en este algoritmo y se comentarán algunos detalles. La ecuación 3.18 tiene dos partes: una es la DFT y la otra el promedio de los 4 LS. La paralelización de la DFT se ha desarrollado en el apartado 3.2.2 y el promedio se puede pensar como un filtro recursivo en donde lo único crítico es el valor del retardo que debe ser múltiplo de 16 para que no se produzcan realimentaciones no deseadas, tal como se ha explicado en el apartado 3.2.1.2 para el caso del filtro de promedio exponencial; en este caso el retardo es de N_{ls} muestras y es múltiplo de 16. La ecuación 3.19 es un cociente entre muestras de dos vectores con el mismo índice, con lo cual, la versión paralela del algoritmo no presenta ninguna dificultad. Por último, la ecuación 3.20 se puede interpretar como un filtro no recursivo lo cual no presenta ningún inconveniente a la hora de paralelizarlo.

3.2.3.2 Compensador de canal

Para llevar adelante la compensación del canal, los símbolos OFDM recibidos son multiplicados por la inversa de la estimación del canal tal como se muestra a continuación:

$$R_c[k] = \frac{R_{uc}[k]}{\hat{H}[k]}, \quad (3.21)$$

donde $R_{uc}[k]$ y $R_c[k]$ se corresponden con los símbolos OFDM (en el dominio de la frecuencia) sin compensar y compensados, respectivamente, y donde k es el índice de la subportadora. Para evitar la división entre números complejos la ecuación 3.21 puede ser redefinida como un producto de dos números complejos utilizando la inversa de la estimación del canal, dando como resultado la ecuación 3.22.

$$R_c[k] = R_{uc}[k] \cdot \hat{H}[k]^{-1} \quad (3.22)$$

Y la inversión del canal estimado se puede obtener de la siguiente manera:

$$\hat{H}[k]^{-1} = \frac{LS_t[k]}{LS_{rm}[k]} = \frac{1}{|LS_{rm}[k]|^2} \cdot LS_t[k] \cdot LS_{rm}^*[k], \quad (3.23)$$

donde $LS_{rm}^*[k]$ es el complejo conjugado de $LS_{rm}[k]$ y gracias a que los valores de $|LS_{rm}[k]|^2$ se pueden acotar a un rango de valores con una determinada precisión, la inversión $1/|LS_{rm}[k]|^2$ se puede realizar mediante una tabla. Por último, recordar que el LS tiene sus subportadoras moduladas con BPSK, por lo tanto, en la ecuación anterior $LS_t[k]$ sólo puede valer ± 1 .

La paralelización de este algoritmo no presenta ninguna dificultad añadida.

3.2.3.3 Modelo de precisión finita

En el caso de los algoritmos en los que a priori no se pueden conocer cuáles serán los rangos numéricos de salida de las diferentes etapas se corre el riesgo de diseñar un sistema con mayor precisión y rango de lo realmente necesario, lo que implicaría un sobre coste de *hardware*. Es por esto que la mejor opción en estos casos es realizar una simulación que se acerque lo más posible al entorno en el cual será usado el algoritmo para poder analizar cuáles son los rangos numéricos de las señales internas del ecualizador y, posteriormente, proceder a una valoración de la precisión numérica de cada una de las etapas del algoritmo.

Utilizando el simulador del sistema OFDM óptico descrito en el apartado 1.4 se ha procedido en primera instancia a valorar los valores máximos y mínimos (rango numérico) de cada una de las señales internas del ecualizador para diferentes escenarios (EBTB, OBTB, fibra). De esta manera se determina la cantidad de bits necesarios para la parte entera de cada una de las señales. Una vez hecho esto se procede a evaluar la parte fraccionaria (precisión) de las señales y el uso o no de algún tipo de redondeo, teniendo muy presente los recursos disponible en la FPGA a utilizar.

En la primera etapa del estimador de canal se obtiene el módulo de la señal compleja $LS_{rm}[k]$, la cual fue calculada anteriormente en el bloque FFT cuya salida es de 20 bits con signo (s11.9). Mediante la simulación se ha obtenido que no todos los 20 bits son necesarios para el cálculo de la estimación del canal, de hecho el rango dinámico de $LS_{rm}[k]$ se cubre con 6 bits manteniendo la misma precisión (s6.9). Obtener el módulo de un valor complejo, que se representa mediante 15 bits para la parte real y otros 15 bits para la parte fraccionaria, implica utilizar un formato de salida de 30 bits (u12.18) calculados de forma teórica. Gracias a la simulación, este valor se puede acotar a tan sólo 12 bits (u10.2) con redondeo y esta reducción provoca un ahorro significativo en el

tamaño de la memoria que se utiliza para obtener la inversa de dicho valor. De esta manera, son necesarias tan sólo 4098 (2^{12}) posiciones de memoria para almacenar todos los posibles valores que pueda adoptar el módulo. También surge de la simulación que la inversa de este módulo es siempre menor a 1 y mayor a 0, por lo tanto, se puede utilizar un formato sin signo y sin parte entera, quedando como incógnita cuántos bits son necesarios para la parte fraccionaria. Teniendo en cuenta que la siguiente operación en el diagrama de bloques del estimador es una multiplicación, se ha valorado la opción de utilizar 18 bits (u0.18) para la parte fraccionaria y se ha comprobado que es suficiente en el simulador.

Hasta aquí el análisis del primer término de la ecuación 3.23. El segundo término se corresponde con los valores de las subportadoras (BPSK) del LS transmitido que se pueden representar con un sólo bit (± 1). Por último, encontramos el conjugado del promedio de los LS recibidos en el dominio de la frecuencia, al cual se le asigna el mismo rango y precisión numérica que a la salida de la FFT (s6.9). Ya se han obtenido los modelos de precisión finita para los 3 términos que intervienen en la ecuación 3.23 (u0.18, s1.0 y s6.9) lo que da como resultado, en el caso más conservador, 33 bits con signo (s6.27). Nuevamente, una vez finalizada la simulación, se ve que el rango y la precisión están acotados a tan solo 2 bits para la parte entera y 16 bits para la parte fraccionaria (s2.16).

Antes de pasar a la etapa de compensación del canal toca hablar del interpolador lineal que se encuentra a la salida del estimador de canal. Si bien de la ecuación 3.20 se puede concluir que existe la posibilidad de que la parte entera se incremente en un bit, se debe recordar que posteriormente se aplica un factor de escala de 0,5 ($1/\sqrt{4}$) lo cual hace que tanto la entrada como la salida del interpolador tengan el mismo rango y precisión numérica (s2.16). Para realizar este truncamiento de la parte fraccionaria se ha utilizado redondeo.

Por último, se encuentra la etapa de compensación del canal (ver ecuación 3.22), la cual realiza la multiplicación entre dos señales (s6.9 y s2.16) y que una vez realizada la simulación se puede acotar el producto resultante a 25 bits (s6.19). La decisión de utilizar 25 bits está relacionada con que en el siguiente bloque (desmapeador) del flujo de procesamiento se encontraran unos multiplicadores *hardware*.

3.2.3.4 Simulación

Utilizando el simulador del apartado 1.4 se probaron diferentes números de repeticiones de LS en el preámbulo y diferentes factores de diezmado para la interpolación. En la Figura 3.26 se presentan dos grupos de curvas de BER en función de la SNR, para diferentes números de repeticiones del LS pero con diezmado constante igual a 4 y para diferentes valores de diezmado con un tamaño total de 1024 muestras. En el primer caso (Figura 3.26.a) se puede ver que al aumentar el número de repeticiones, o lo que es lo mismo, al aumentar el número de LSs que se promedian en el tiempo se logra disminuir la diferencia de BER con respecto al máximo teórico hasta un valor de $0,7 \times 10^{-3}$ para el caso de 16 repeticiones. En el caso de la Figura 3.26.b se puede ver que al aumentar el número de repeticiones y diezmado, pero manteniendo un tamaño final constante, se logra disminuir la diferencia de BER con respecto al máximo teórico hasta un valor de $0,5 \times 10^{-3}$ para el caso de 16 repeticiones de 64 muestras longitud.

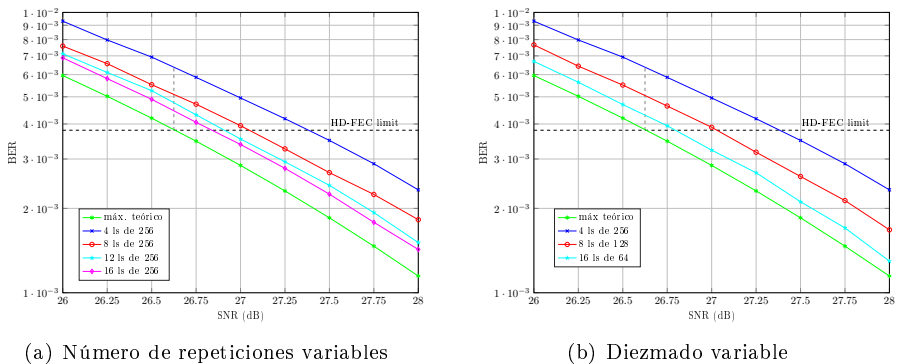


Figura 3.26: BER vs SNR para diferentes número de diezmado y repeticiones, para 512 subportadoras moduladas con 256 QAM.

Aumentar el número de repeticiones implica utilizar un preámbulo más grande, con lo cual disminuye un poco la tasa binaria, y utilizar mayores recursos de *hardware* para implementar el filtro de promediado. Aumentar la cantidad de diezmado aplicada trae como consecuencia que un mayor número de subportadoras de baja frecuencia no puedan ser utilizadas, hay que tener en cuenta que estas suelen tener los mejores valores de EVM.

Para poder aprovechar al máximo las subportadoras de baja frecuencia se ha utilizado un factor de diezmado por 4 y para reducir el coste *hardware* se ha

decidido trabajar con tan sólo 4 repeticiones del LS, lo cual no afectaba de manera significativa a la BER del sistema (error $< 2,5 \times 10^{-3}$) para diferentes escenarios.

3.2.4 Desmapeo

El proceso de mapeo asigna a una combinación de bits un punto en la constelación, luego divide ese punto por un factor de escala y se lo asigna a una subportadora OFDM. Las Figuras 3.27 y 3.28 muestran el mapeo Gray para las diferentes constelaciones y los factores de escala asociados utilizados en este trabajo. Los símbolos están normalizados de modo que cada constelación tiene la misma potencia media.

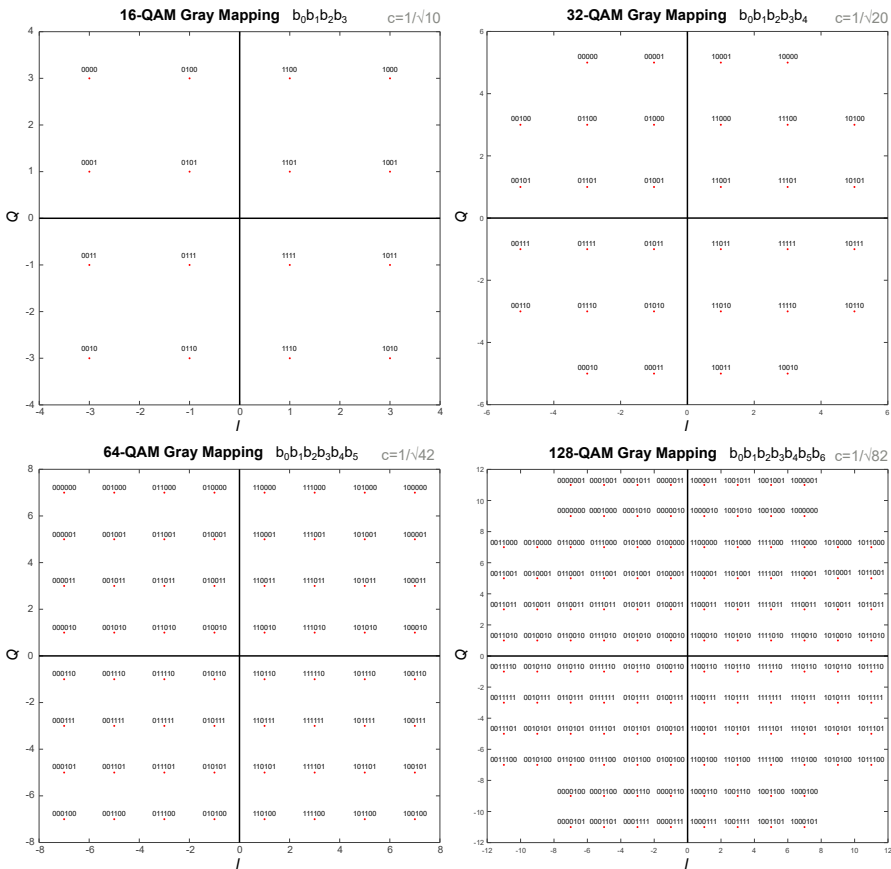


Figura 3.27: Constelaciones 16-QAM, 32-QAM, 64-QAM y 128-QAM con mapeo Gray.

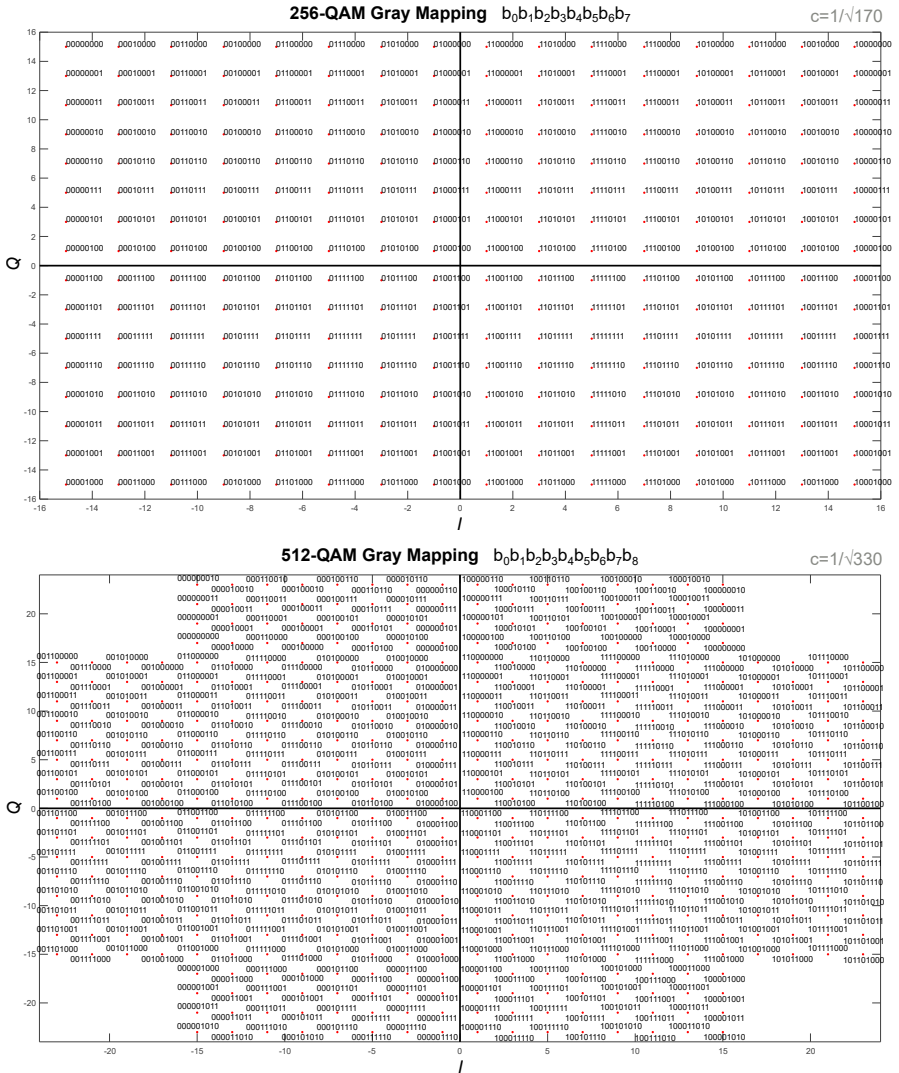


Figura 3.28: Constelaciones 256-QAM y 512-QAM con mapeo Gray.

El desmapeador realiza la operación complementaria, primero multiplica por el factor de escala, a continuación decide a qué punto de la constelación M-QAM corresponde y, por último, extrae el flujo de bits de la secuencia compleja recibida en cada subportadora. Un determinado valor de magnitud y fase recibidos representan una cierta combinación de bits dependiendo de la constelación uti-

lizada para cada subportadora. El orden de modulación (M-QAM) para cada subportadora se ha determinado mediante el algoritmo de *bitloading* que, en función de la SNR, maximiza la tasa binaria manteniendo una tasa de error por debajo del límite del corrector de errores.

En el receptor se extrae la parte real e imaginaria de cada subportadora y se debe tomar una decisión acerca de qué combinación de bits envió el transmisor. Debido a que cada una de las subportadoras han sido distorsionadas por el canal de transmisión, hay un error entre la posición recibida y transmitida de cada punto de la constelación. Para decidir sobre a qué punto de la constelación se debe asignar el valor recibido, el desmapeador utilizado en esta tesis no trabaja con el cálculo del error, sino que divide el plano complejo en regiones de igual tamaño, una por cada bit de salida (para el caso de constelaciones cuadradas), y en función a qué regiones pertenece la señal recibida ($r = \alpha + j\beta$) genera la combinación de bits correspondiente. A este método se lo conoce como desmapeador *bit-level* [129].

Para entender cómo trabaja el desmapeador *bit-level* analicemos el caso para 16-QAM a modo de ejemplo. En la Figura 3.29 se observan las 4 regiones de desmapeo, una por cada bit, 2 de ellas (b_0 y b_1) dependen únicamente de los valores reales (en fase) de la señal recibida y las otras dos (b_2 y b_3) de la parte imaginaria (en cuadratura). Si analizamos la región correspondiente al bit 0 (arriba a la izquierda) vemos que si la parte real de la señal recibida es positiva este bit valdrá 1 a la salida del desmapeador y si cae fuera de esta región valdrá 0. Analizando esta y las restantes regiones se pueden obtener el siguiente conjunto de ecuaciones:

$$\begin{aligned} b_0 &= \begin{cases} 1, \text{Si } \alpha \geq 0 \\ 0, \text{resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{Si } |\alpha| \leq 2 \\ 0, \text{resto de casos} \end{cases} \\ b_2 &= \begin{cases} 1, \text{Si } \beta \leq 0 \\ 0, \text{resto de casos} \end{cases} & b_3 &= \begin{cases} 1, \text{Si } |\beta| \leq 2 \\ 0, \text{resto de casos} \end{cases} \end{aligned} \quad (3.24)$$

De la ecuación 3.24 se infiere que únicamente son necesarias 6 comparaciones (3 para la parte real y 3 para la parte imaginaria) para el algoritmo de desmapeo. Una particularidad de las regiones de decisión es que las pertenecientes a los bits menos significativos son las mismas regiones que las de los bits más significativos rotadas 90° . Este algoritmo se puede generalizar para diferentes constelaciones cuadradas tal como se ha hecho en [129] y obtener así el conjunto de ecuaciones para las constelaciones utilizadas en este trabajo. En

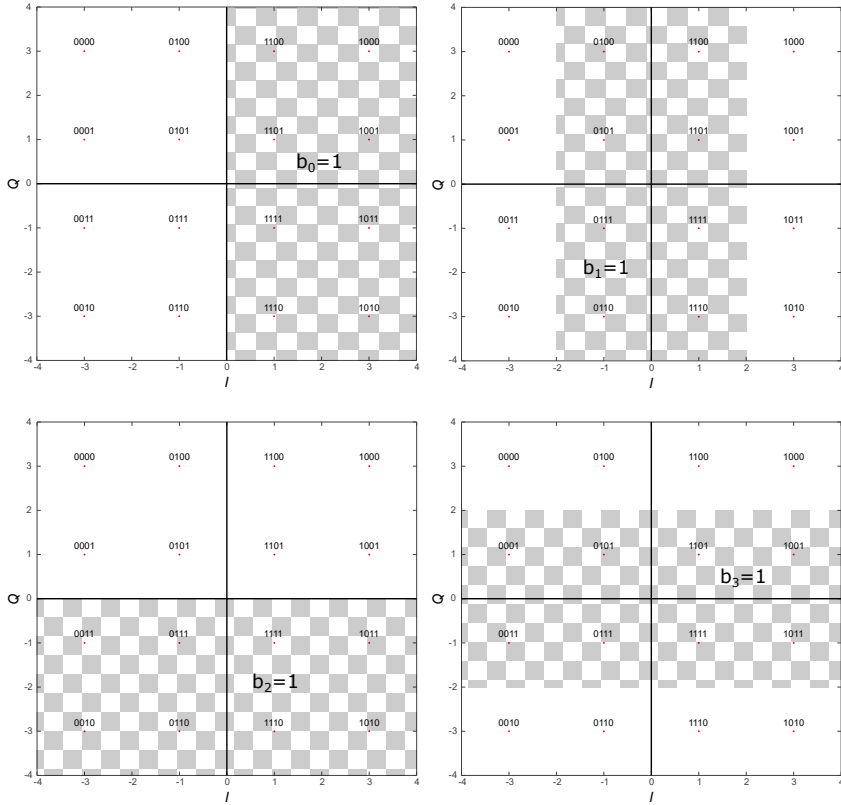


Figura 3.29: Regiones de desmapeo para el caso de 16-QAM.

las ecuaciones 3.25 y 3.26 se presentan las reglas de decisión para el caso de 64-QAM y 256-QAM respectivamente.

$$\begin{aligned}
 b_0 &= \begin{cases} 1, \text{ Si } \alpha \geq 0 \\ 0, \text{ resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{ Si } |\alpha| \leq 4 \\ 0, \text{ resto de casos} \end{cases} & b_2 &= \begin{cases} 1, \text{ Si } 2 \leq |\alpha| \leq 6 \\ 0, \text{ resto de casos} \end{cases} \\
 b_3 &= \begin{cases} 1, \text{ Si } \beta \leq 0 \\ 0, \text{ resto de casos} \end{cases} & b_4 &= \begin{cases} 1, \text{ Si } |\beta| \leq 4 \\ 0, \text{ resto de casos} \end{cases} & b_5 &= \begin{cases} 1, \text{ Si } 2 \leq |\beta| \leq 6 \\ 0, \text{ resto de casos} \end{cases}
 \end{aligned} \tag{3.25}$$

$$\begin{aligned}
 b_0 &= \begin{cases} 1, \text{ Si } \alpha \geq 0 \\ 0, \text{ resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{ Si } |\alpha| \leq 8 \\ 0, \text{ resto de casos} \end{cases} \\
 b_2 &= \begin{cases} 1, \text{ Si } 4 \leq |\alpha| \leq 12 \\ 0, \text{ resto de casos} \end{cases} & b_3 &= \begin{cases} 1, \text{ Si } 2 \leq |\alpha| \leq 6 \text{ ó } 10 \leq |\alpha| \leq 14 \\ 0, \text{ resto de casos} \end{cases} \\
 b_4 &= \begin{cases} 1, \text{ Si } \beta \leq 0 \\ 0, \text{ resto de casos} \end{cases} & b_5 &= \begin{cases} 1, \text{ Si } |\beta| \leq 8 \\ 0, \text{ resto de casos} \end{cases} \\
 b_6 &= \begin{cases} 1, \text{ Si } 4 \leq |\beta| \leq 12 \\ 0, \text{ resto de casos} \end{cases} & b_7 &= \begin{cases} 1, \text{ Si } 2 \leq |\beta| \leq 6 \text{ ó } 10 \leq |\alpha| \leq 14 \\ 0, \text{ resto de casos} \end{cases}
 \end{aligned} \tag{3.26}$$

Para constelaciones no rectangulares, las formas y simetrías de las regiones cambian un poco; a modo de ejemplo veamos el caso de 32-QAM. En la Figura 3.30 se presentan las regiones de decisión de los 5 bits de salida del desmapeador para el caso de 32 QAM. A diferencia del caso de constelaciones cuadradas, aquí las regiones de los bits menos significativos no son las regiones de los bits más significativos rotadas 90°, sino que aparecen nuevas formas de regiones en las que intervienen tanto la parte imaginaria como la real.

De la misma manera que se hizo para constelaciones rectangulares, se pueden analizar las regiones de toma de decisión para cada uno de los bits de salida del desmapeador y expresarlas mediante un conjunto de ecuaciones. Las reglas de decisión para el caso de 32-QAM, 128-QAM y 512-QAM se describen en las ecuaciones 3.27, 3.28 y 3.29 respectivamente.

$$\begin{aligned}
 b_0 &= \begin{cases} 1, \text{ Si } \alpha \geq 0 \\ 0, \text{ resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{ Si } |\alpha| \leq 4 \text{ y } |\beta| \leq 4 \\ 0, \text{ resto de casos} \end{cases} & b_2 &= \begin{cases} 1, \text{ Si } |\alpha| \geq 2 \text{ y } |\beta| \leq 4 \\ 0, \text{ resto de casos} \end{cases} \\
 b_3 &= \begin{cases} 1, \text{ Si } \beta \leq 0 \\ 0, \text{ resto de casos} \end{cases} & b_4 &= \begin{cases} 1, \text{ Si } |\beta| \leq 2 \text{ ó } |\alpha| \leq 2 \text{ y } |\beta| \geq 4 \\ 0, \text{ resto de casos} \end{cases}
 \end{aligned} \tag{3.27}$$

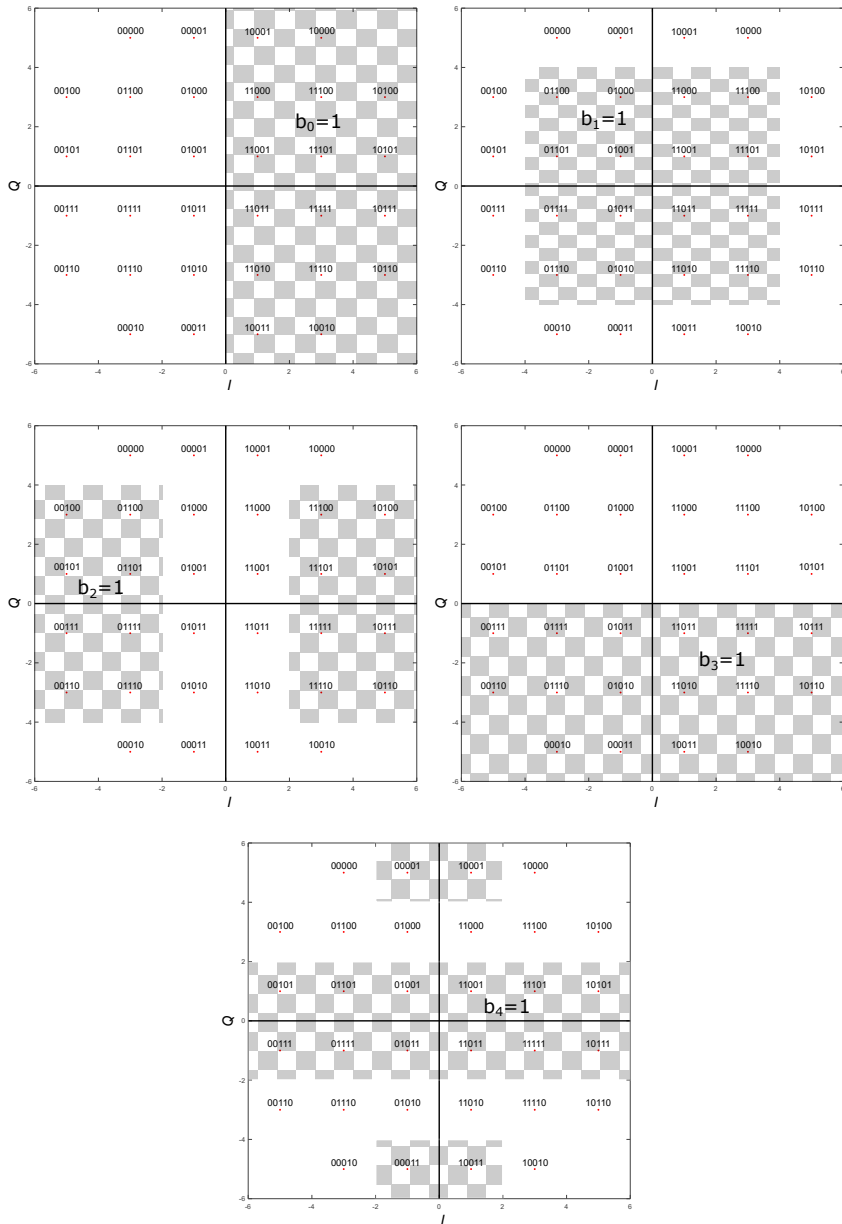


Figura 3.30: Regiones de desmapeo para el caso de 32-QAM.

$$\begin{aligned}
 b_0 &= \begin{cases} 1, \text{ Si } \alpha \geq 0 \\ 0, \text{ resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{ Si } |\alpha| \leq 8 \text{ y } |\beta| \leq 8 \\ 0, \text{ resto de casos} \end{cases} & b_2 &= \begin{cases} 1, \text{ Si } |\alpha| \geq 4 \text{ y } |\beta| \leq 8 \\ 0, \text{ resto de casos} \end{cases} \\
 b_3 &= \begin{cases} 1, \text{ Si } 2 \leq |\alpha| \leq 6 \text{ ó } |\alpha| \geq 10 \\ 0, \text{ resto de casos} \end{cases} & b_4 &= \begin{cases} 1, \text{ Si } \beta \leq 0 \\ 0, \text{ resto de casos} \end{cases} \\
 b_5 &= \begin{cases} 1, \text{ Si } |\beta| \leq 4 \text{ ó } |\alpha| \leq 4 \text{ y } |\beta| \geq 8 \\ 0, \text{ resto de casos} \end{cases} & b_6 &= \begin{cases} 1, \text{ Si } 2 \leq |\beta| \leq 6 \text{ ó } |\beta| \geq 10 \\ 0, \text{ resto de casos} \end{cases}
 \end{aligned} \tag{3.28}$$

$$\begin{aligned}
 b_0 &= \begin{cases} 1, \text{ Si } \alpha \geq 0 \\ 0, \text{ resto de casos} \end{cases} & b_1 &= \begin{cases} 1, \text{ Si } |\alpha| \leq 16 \text{ y } |\beta| \leq 16 \\ 0, \text{ resto de casos} \end{cases} & b_2 &= \begin{cases} 1, \text{ Si } |\alpha| \geq 8 \text{ y } |\beta| \leq 16 \\ 0, \text{ resto de casos} \end{cases} \\
 b_3 &= \begin{cases} 1, \text{ Si } 4 \leq |\alpha| \leq 12 \text{ ó } |\alpha| \geq 20 \\ 0, \text{ resto de casos} \end{cases} & b_4 &= \begin{cases} 1, \text{ Si } 2 \leq |\alpha| \leq 6 \text{ ó } 10 \leq |\alpha| \leq 14 \text{ ó } 18 \leq |\alpha| \leq 22 \\ 0, \text{ resto de casos} \end{cases} \\
 b_5 &= \begin{cases} 1, \text{ Si } \beta \leq 0 \\ 0, \text{ resto de casos} \end{cases} & b_6 &= \begin{cases} 1, \text{ Si } |\beta| \leq 8 \text{ ó } |\alpha| \leq 8 \text{ y } |\beta| \geq 16 \\ 0, \text{ resto de casos} \end{cases} \\
 b_7 &= \begin{cases} 1, \text{ Si } 4 \leq |\beta| \leq 12 \text{ ó } |\beta| \geq 20 \\ 0, \text{ resto de casos} \end{cases} & b_8 &= \begin{cases} 1, \text{ Si } 2 \leq |\beta| \leq 6 \text{ ó } 10 \leq |\beta| \leq 14 \text{ ó } 18 \leq |\beta| \leq 22 \\ 0, \text{ resto de casos} \end{cases}
 \end{aligned} \tag{3.29}$$

De las ecuaciones presentadas en este apartado se desprende que la complejidad computacional medida en cantidad de comparaciones para las diferentes constelaciones es la que se muestra en la Tabla 3.6, que hay ciertas comparaciones que son comunes a los bits de otras constelaciones (por ejemplo $\alpha \geq 0$ o $\beta \leq 0$) y que también existen algunas que son complementarias entre sí (por ejemplo $|\alpha| \leq 4$ y $|\alpha| \geq 4$). Si se analizan las comparaciones necesarias para implementar el desmapeador M -QAM, se obtiene que sólo son necesarios algunos de los comparadores correspondientes a la constelación de mayor orden para implementar las de menor orden. Esto quiere decir que para nuestro caso sólo serán necesarios 46 comparadores (23 para la parte real y 23 para la parte imaginaria) para el bloque de desmapeo capaz de trabajar con constelaciones que van desde 16-QAM hasta 512-QAM.

Este algoritmo es muy simple, rápido y no degrada el rendimiento de la BER, tanto para canales AWGN como para canales con desvanecimiento. Gracias a su topología, su implementación en paralelo y segmentada no presenta mayores

Tabla 3.6: Complejidad *hardware* del desmapeador M-QAM

	16 QAM	32 QAM	64 QAM	128 QAM	256 QAM	512 QAM	1024 QAM
Comparadores	6	10	14	20	30	46	62

dificultades. Por último, decir que para procesar 16 muestras de subportadoras, de manera simultánea, es necesario replicar 16 veces este algoritmo.

3.2.4.1 Modelo de precisión finita

En la primera etapa del desmapeador se realiza el escalado del dato recibido, lo cual consiste en multiplicar la señal recibida por un valor, que depende del orden de modulación utilizado. Para evaluar la precisión de esta etapa es conveniente tener en cuenta las limitaciones que puede imponer el *hardware*. En la FPGA Virtex-7 se disponen de multiplicadores por *hardware* que cuentan con 25 bits para el multiplicando, 18 bits para el multiplicador y 43 bits para el producto. La salida del bloque anterior (ecualizador) es de 25 bits, lo cual nos deja 18 bits para la constante de escalado. Los factores de escala van desde 3,1623 para el caso de 16 QAM hasta 18,1659 para 512 QAM, lo que determina la necesidad de utilizar 5 bits sin signo para la parte entera ($2^5 = 32$) dejando los 13 bits restantes para la parte fraccionaria (u5.13).

El formato de salida del ecualizador es de 25 bits con signo, 6 para la parte entera y 19 bits para la parte fraccionaria (s6.19), tanto para la parte real como imaginaria. Después de multiplicar por la constante de escalado (u5.13), el formato de salida es de 43 bits con signo, 11 para la parte entera y 32 bits para la parte fraccionaria (s11.32). Sin embargo, el valor máximo de la parte real o imaginaria para la constelación más grande es de ± 23 , con lo cual bastaría con utilizar 6 bits con signo para la parte entera (± 32) y conservar los 32 bits para la parte fraccionaria (s6.32). Esto deja un margen bastante amplio para que la distorsión que haya introducido el canal de transmisión en la señal transmitida esté correctamente representada en este formato.

La segunda etapa corresponde a determinar a qué regiones pertenece el valor escalado recibido según las ecuaciones 3.24, 3.27, 3.25, 3.28, 3.26 y 3.29. Antes de continuar con el estudio de precisión finita de esta etapa, analizaremos brevemente el caso particular de las fronteras de las regiones de la constelación 32 QAM (ver Figura 3.30). Si se plantea el caso de haber recibido un valor

$\beta = 2$ y un valor α cualquiera, se puede ver que este valor r ($\alpha + j\beta$) siempre se encuentra a la misma distancia vertical de dos puntos válidos de la constelación, por lo tanto, el b_4 podría valer tanto 0 como 1, y en ambos casos, esto sería correcto; lo mismo ocurre para $\beta = -2$. Esta ambigüedad no se puede admitir y si se observa la ecuación 3.27 se ve que para los valores $|\beta| = 2$, b_4 toma el valor de 1. Esta decisión, desde el punto de vista del *hardware*, no es la más adecuada porque utiliza más recursos que en el caso de $\beta < 2$ y $\beta \geq -2$, b_4 valiese 1. Para ilustrar este problema y su solución, a continuación se cuantifican los valores de β con 4 bits para la parte entera y 2 bits para la parte fraccionaria (s4.2) y se analiza el entorno de la frontera de $|\beta| = 2$:

Decimal	-2.25	-2.00	-1.75	1.50	...	-0.25	0.00	0.25	...	1.50	1.75	2.00	2.25
Binario	1101.11	1110.00	1110.01	1110.10	...	1111.11	0000.00	0000.01	...	0001.10	0001.11	0010.00	0010.01

Analizando este ejemplo se pueden extraer dos conclusiones muy importantes que se pueden aplicar a todas las fronteras de todas las modulaciones:

- Basta con analizar la parte entera de la señal recibida para determinar el valor del bit de salida del desmapeador.
- Si se reemplaza en las ecuaciones de asignación de bits el símbolo \leq por $<$ para los valores positivos de α y β , se resuelve el conflicto de frontera con un menor coste computacional sin introducir error alguno.

Tomando estas dos premisas como válidas se puede reescribir la ecuación del b_4 para 32 QAM de la siguiente manera:

$$b_4 = \begin{cases} 1, \text{ Si } -2 \leq \beta < 2 \\ 1, \text{ Si } -2 \leq \alpha < 2 \text{ y } \beta \geq 4 \\ 1, \text{ Si } -2 \leq \alpha < 2 \text{ y } \beta < -4 \\ 0, \text{ resto de casos} \end{cases}$$

De esta manera se resuelve el problema de las fronteras de las regiones que será aplicado a todas las ecuaciones de desmapeo (3.24, 3.27, 3.25, 3.28, 3.26 y 3.29) y, ahora sí, es el momento de continuar con el análisis del modelo de precisión finita.

La señal escalada tiene un ancho de palabra de 38 bits con signo, tanto para la parte real como imaginaria, de los cuales 6 son para la parte entera y 32 bits para la parte fraccionaria (s6.32). Como se ha dicho anteriormente, para

determinar a qué región pertenece la señal recibida sólo será necesaria la parte entera de la misma sin ningún tipo de redondeo previo, esto implica que el ancho de palabra para las entradas de los comparadores será de 6 bits con signo (s6.0). El ancho de palabra a la salida del desmapeador viene dado por el orden de modulación, por ejemplo 4 bits para 16 QAM y 9 bits para 512 QAM.

3.2.4.2 Simulación

Se ha desarrollado en MATLAB un modelo de punto fijo del desmapeador obviando la parte del escalado. Como algoritmo de referencia se ha escogido el demodulador QAM para constelaciones rectangulares perteneciente al paquete de sistema de comunicaciones de MATLAB. Para verificar su comportamiento se han generado dos señales aleatorias ($\alpha + j\beta$) de doble precisión en un intervalo de ± 32 , y para el modelo de punto fijo estas señales fueron convertidas a enteros descartando su parte fraccionaria (*floor*). Como resultado de la simulación se han obtenido que, las salidas de ambos algoritmos para el caso de las constelaciones cuadradas son idénticas (0 errores), mientras que en el caso de constelaciones rectangulares sólo se han observado diferencias en las esquinas de la constelación. Esto último no debe ser considerado un error ya que es un caso particular en el cual el valor recibido puede ser asignado a múltiples puntos de la constelación.

3.2.5 Simulación

En primera instancia se ha desarrollado un modelo en MATLAB para el transmisor y receptor en punto flotante de 64 bits que ha permitido validar el funcionamiento de los algoritmos y modelar las respuestas reales de los elementos que componen la etapa analógica y óptica. Se han realizado pruebas con diferentes tipos de preámbulos para valorar diversos algoritmos de sincronización temporal y de estimación y compensación de canal. Luego se han evaluado diferentes técnicas (*clipping*, *bitloading* y preénfasis) para mejorar las prestaciones del sistema teniendo en cuenta la caracterización del sistema OFDM óptico obtenida en el capítulo 2. Una vez seleccionados los algoritmos a utilizar, se han codificado los modelos de precisión finita de cada uno de los algoritmos y se ha evaluado cómo afecta esto a las prestaciones del sistema.

Para verificar el correcto funcionamiento del receptor y de sus prestaciones, se han utilizados 3 tipos de señales OFDM. La primera (tipo I) se corresponde a una señal sintética donde se anulan los efectos de todos los componentes analógicos y ópticos presentes en el sistema OFDM. La segunda (tipo II),

también sintética, se han tenido en cuenta dichos efectos. Y la tercera (tipo III) se corresponde con señales reales capturadas con el conversor AD y generadas con el transmisor OFDM descrito en el apartado 3.1 para distintos esquemas de medición (EBTB, OBTB, SSMF). Se han realizado comprobaciones bit a bit de los datos de salida del receptor para los diferentes esquemas y de esta manera se ha validado el correcto funcionamiento del diseño en precisión finita del receptor OFDM. A continuación, en la Figura 3.31 se muestran las constelaciones obtenidas para los tres tipos de señales antes mencionados, para el caso de los algoritmos con doble precisión y con precisión finita.

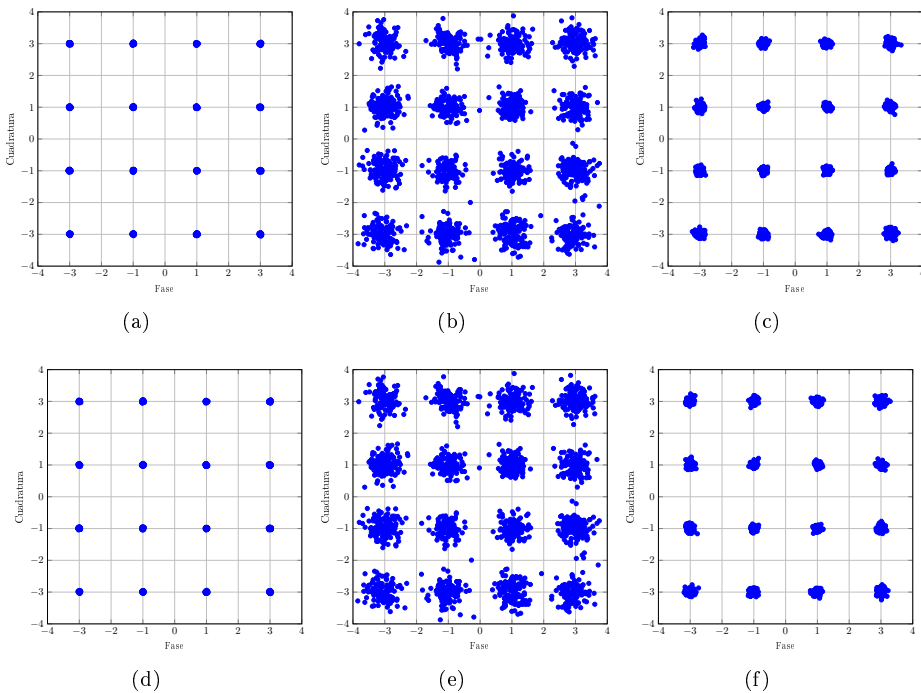


Figura 3.31: Diagramas de constelación 16-QAM para distintos tipos de señales utilizando algoritmos sin y con precisión finita. (a) y (d) tipo I (EVM=0,45 %), (b) y (e) tipo II (EVM=12 %), y (c) y (f) tipo III (EVM=2,9 %).

En la Figura 3.31 no se observan diferencias entre las constelaciones obtenidas con los algoritmos en doble precisión y sus versiones de precisión finita para ninguno de los tres tipos de señales, lo cual confirma que los rangos dinámicos y la precisión utilizadas en cada una de las etapas del diseño son adecuados. A estas constelaciones se les ha medido el EVM promedio y no se han obte-

nido diferencias significativas ($< 0,1\%$) producto de la precisión finita de los algoritmos. Como se menciona en [41], el EVM es una buena métrica para medir las prestaciones de un sistema OFDM y tiene una relación directa con la SNR y BER del sistema. También se realizaron pruebas con subportadoras cargadas con distintos índices de modulación (*bit loading*) para diferentes escenarios (EBTB, OBTB, SSMF) y no se han observado diferencias en el EVM promedio ni en el EVM máximo y mínimo.

3.3 Conclusiones

En este capítulo, se ha desarrollado en MATLAB un modelo funcional (secuencial y en coma flotante) y un modelo *hardware* (codificación con funciones básicas y en punto fijo) del transmisor y receptor, así como también se ha modelado el comportamiento de los conversores y la etapa óptica. Estos modelos han permitido seleccionar y verificar los algoritmos empleados, y serán utilizados más adelante para generar los estímulos/resultados para la verificación de la implementación en FPGA.

En lo que concierne al transmisor, se han podido evaluar diferentes técnicas para mejorar las prestaciones del mismo y determinar valores de partida, que deberán ser optimizados posteriormente mediante pruebas en el sistema óptico de tiempo real que se describe en el capítulo 5.

Se ha propuesto un nuevo algoritmo de sincronización temporal para sistemas OFDM ópticos que utilizan detección directa. Este algoritmo saca provecho del uso de un preámbulo repetitivo, donde el receptor realiza una correlación cruzada entre la señal recibida y la parte repetitiva del preámbulo usando sólo un bit para cuantificar ambas señales. El uso de un preámbulo repetitivo ha permitido reducir la longitud de la correlación cruzada y, por lo tanto, reducir también el coste computacional del mismo que, en comparación con el resto de los algoritmos estudiados, es mucho menor. En [130] se ha presentado el diseño e implementación en FPGA y se ha validado experimentalmente en un banco de pruebas de comunicación óptica, demostrando la viabilidad de algoritmo y sus buenas prestaciones en escenarios con baja SNR.

En este capítulo se han presentado dos diseños, longitud fija y longitud variable, de un procesador FFT de alta velocidad y el diseño de los módulos de reordenamiento de datos necesarios para que los muestras entren y salgan del sistema en orden natural. El algoritmo FFT utilizado es *radix-4* con diezmo en frecuencia y ha sido diseñado mediante una arquitectura paralela segmentada que utiliza conmutadores de retardos multicaminos. Este módulo procesa

16 muestras simultáneamente para lo cual utiliza 4 mariposas R4MDC en paralelo por etapa. Esta topología ha permitido eliminar el primer conmutador inter-etapa, lo cual ha favorecido a la latencia del sistema. El diseño de longitud fija frente al de longitud variable tiene la ventaja de utilizar una cantidad de recursos que se ajusta la longitud de la FFT y de necesitar módulos de ordenamiento de datos con un menor coste *hardware*. La latencia de ambos módulos es la misma y varía según la longitud seleccionada.

Para la propuesta del ecualizador de canal ha tenido en cuenta que, en los sistemas ópticos alámbricos los canales pueden ser considerados invariables en el tiempo, razón por la cual, se ha escogido una disposición *block-type* para las subportadoras a utilizar en la estimación del canal. Esta hipótesis se deberá comprobar en la etapa experimental. Para no aumentar el tamaño del preámbulo, se ha utilizado un diezmado por 4 del LS, se ha escogido una interpolación lineal y se lo ha replicado 4 veces para disminuir el ruido de la estimación.

Sobre el desmapeador, simplemente decir que el algoritmo utilizado es lo que se denomina desmapeador *bit-level* y que este analiza los bits del símbolo recibido para determinar a qué región de decisión pertenece y así asignar a qué punto de la constelación QAM se corresponde.

Por último, decir que en este diseño se han tenido en cuenta los objetivos principales de esta tesis: se han desarrollado algoritmos que pueden procesar 16 muestras en paralelo, que pueden ser segmentados con facilidad y que a la hora de establecer el sistema de punto fijo utilizado se han valorado los recursos *hardware* disponibles en la FPGA.

Implementación hardware del módem OFDM óptico

En este capítulo se detallan las decisiones de implementación y los recursos de la FPGA que se han utilizado para llevar a cabo tanto el transmisor como el receptor OFDM, como así también los bancos de pruebas utilizados para verificar el correcto funcionamiento de todo el sistema. Para ambos diseños se ha utilizado la tarjeta de evaluación VC707 de Xilinx que cuenta con un dispositivo FPGA Virtex-7 cuya frecuencia máxima de operación alcanza los 540 MHz.

Al respecto de los conversores decir que ambos alcanzan una tasa de muestreo de hasta 5 GS/s y que poseen una interfaz de datos LVDS paralela que presentan varios desafíos e inconvenientes que se han de resolver desde la FPGA. Ambos conversores trabajan con 4 muestras en paralelo lo que implica que la FPGA debe ser capaz de procesar 4 muestras a una frecuencia de 1,25 GHz, lo cual no es viable y hace necesario la deserialización/serialización de las muestras según sea el caso del ADC o DAC. Utilizando un factor de 4 para la serialización/deserialización se ha logrado reducir la frecuencia de trabajo de la FPGA hasta los 312,5 MHz, lo cual trae implícito un incremento del área utilizada, producto de la necesidad de tener que procesar 16 muestras en paralelo.

La tasa binaria que puede alcanzar este sistema depende en gran medida de la frecuencia de muestro de los conversores y estos están en un momento de expansión. Cuando se dio comienzo a este trabajo de tesis se disponía de conversores de hasta 4 GS/s, luego aparecieron conversores de hasta 5 GS/s y en este último año han salido al mercado conversores de hasta 6 GS/s. Por este motivo, a la hora del diseño *hardware*, se tuvo en cuenta que la velocidad de reloj podría llegar a ser mayor y se ha puesto especial cuidado en la segmentación y el control de *fanout* de señales que suelen ser críticas (reset, enables, control de selección de multiplexores, etc).

En líneas generales ambos diseños son muy modulares y se los puede dividir en tres grandes partes: interfaz con el conversor, procesamiento digital de la señal, e interfaz con el usuario. Cada uno de estos bloques tiene su propio dominio de reloj lo cual añade complejidad al diseño. A continuación se presentan las implementaciones *hardware* de transmisor y receptor OFDM.

4.1 Transmisor OFDM

El transmisor se basa en un generador arbitrario de funciones cuyo contenido de memoria se genera en el ordenador utilizando MATLAB, tal como se ha descrito en el apartado 3.1. El contenido de la memoria puede ser leído, escrito y reproducido desde el ordenador a través de una conexión ethernet con la tarjeta de desarrollo. La capacidad de almacenamiento es de un millón (2^{20}) de muestras de 12 bits enteros sin signo, lo que equivale aproximadamente a unos 1000 símbolos OFDM con $N=1024$.

4.1.1 Interfaz DAC

El conversor digital analógico de alta velocidad MD657B es de 12 bits y recibe 4 muestras simultáneamente a través de 48 pares LVDS en modo DDR (*double-data-rate*) que deben ser síncronas con un reloj ($F_s/8$) que se le provee a la FPGA desde el conversor. Este reloj se genera a partir de otro reloj externo, cuya frecuencia debe ser igual a la tasa de muestreo deseada (F_s), que se conecta directamente al DAC. Para alcanzar esta tasa de datos dentro de la FPGA es necesario utilizar los recursos dedicados del bloque E/S. En la Figura 4.1 se muestra el esquema de conexión del conversor con la FPGA y el diagrama de bloques para la serialización de las muestras y la obtención del reloj.

Cada serializador (OSERDES) recibe un bit de 4 muestras consecutivas en el tiempo, a una frecuencia de 312,5 MHz, y utilizando 12 OSERDES en

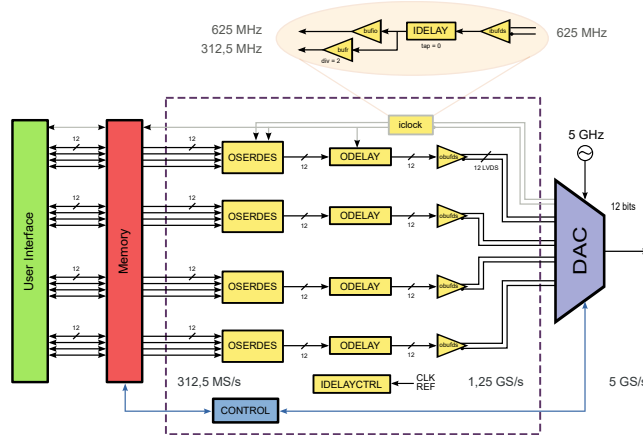


Figura 4.1: Diagrama de bloques de la interfaz con el DAC.

paralelo se obtiene una muestra de 12 bits a una frecuencia de 1,25 GHz. Posteriormente, los 12 bits correspondientes a una misma muestra atraviesan 12 retardos analógicos (ODELAY) y se conectan a 12 búferes diferenciales de salida (OBUFDS). Este esquema se replica 4 veces para obtener 4 muestras en paralelo a una tasa de 1,25 GS/s, lo cual permite que el convertor trabaje a una frecuencia de muestreo de 5 GS/s. En el esquema de la Figura 4.1 se utilizan un total de 48 OSERDESs, 48 ODELAYs y 48 OBUFDSs. Los pines de la FPGA se conectan a través de la tarjeta VC707 hasta un conector de alta densidad (HPC) y de ahí a través de la tarjeta FMC2657 hasta los pines del convertor DA. Al tratarse de tarjetas de evaluación, las 48 señales diferenciales no realizan el mismo recorrido, y a estas frecuencias de operación, provoca que se generen distintos retardos temporales hasta en DAC. La existencia de estos retardos provoca que el convertor interprete o muestree de manera errónea alguna de las 48 señales y esto se traduce en bits erróneos en alguna de las 4 muestras.

Para resolver este inconveniente, el fabricante del convertor, ha incluido un sistema de selección del momento óptimo de muestreo a través de dos bits de selección (SEL1-2), que a continuación se detalla. A partir del reloj externo se generan dentro del convertor 4 relojes con diferentes fases (0° , 90° , 180° , 270°) y a través de las señales SEL1 y SEL2 se escoge cuál de estas 4 fases se ha de utilizar para el muestreo de los 48 pares diferenciales. En la Figura 4.2 se muestra un diagrama de formas de onda con las 4 posibles señales de selección de fase en donde se puede apreciar que el caso de SEL1=1 y SEL2=0 es el indicado para muestrear los bits de las 4 muestras (A, B, C, y D). Esta

solución tiene dos inconvenientes: el primero es que la fase correcta varía según la frecuencia del reloj de muestreo utilizada, y el segundo es que la elección de la fase no se puede automatizar. En el apartado 4.1.4 se explica la solución adoptada para este procedimiento.

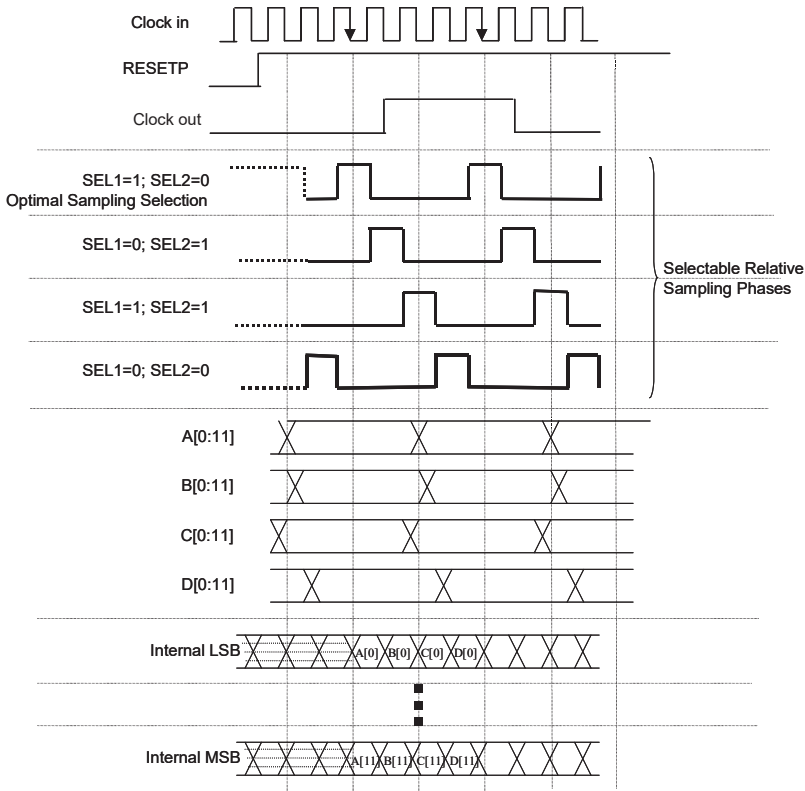


Figura 4.2: Diagrama de formas de onda para la selección del muestreo óptimo de los bits de las muestras del convertor MD657B.

La solución implementada por el fabricante para muestrear correctamente las 48 señales tiene sus limitaciones, como se verá más adelante, el ADC presenta el mismo inconveniente, sin embargo, la solución propuesta por el fabricante es mejor en varios aspectos.

4.1.2 Memoria de almacenamiento

El dispositivo Virtex-7 de la tarjeta VC707 dispone de 1.030 bloques de memoria RAMB36, lo que equivale a unos 37.080 kb de capacidad de almacenamiento máxima. La premisa para este módulo ha sido intentar dotar de la mayor capacidad de almacenamiento posible para las muestras de cada uno de los DACs del AWG. Utilizar la gran mayoría de bloques de memoria impone ciertas limitaciones en la frecuencia por estar ubicados en posiciones específicas dentro del dispositivo.

Valorando las restricciones de frecuencia, un nivel de segmentación no muy excesivo y los posibles problemas de emplazamiento y rutado, se obtuvo que una capacidad de 2^{20} muestras por DAC. Para alcanzar dicha capacidad se han utilizado un total de 384 bloques de memoria de $32k \times 1$ bit de doble puerto sin limitar la frecuencia de operación deseada. Con lo cual, casi el 70 % de los bloques de memoria del dispositivo son utilizados para almacenar las muestras de ambos canales del AWG.

Además del almacenamiento, este módulo contiene lógica de control que sincroniza e interactúa con la interfaz de usuario (4 muestras de 12 bits cada una) y con la interfaz de los DACs (16 muestras de 12 bits cada una). Ambas interfaces utilizan la misma fuente de reloj, en lo que respecta a la memoria de almacenamiento, pero uno es 4 veces más lento que el otro. Aquí se optó por no trabajar con habilitación de reloj sino con dos dominios de reloj diferentes y de ahí el uso de las memorias de doble puerto con dos relojes independientes.

4.1.3 Control e Interfaz de usuario

Esta interfaz funciona desde MATLAB a través de una conexión gigabit ethernet utilizando el protocolo UDP. El core de ethernet utilizado en esta tesis ha sido desarrollado por el laboratorio de comunicaciones digitales (DiComLab) del campus de Gandia de la UPV y ha sido adaptado para esta aplicación en particular. Desde el ordenador se puede escribir, leer y reproducir el contenido de la memoria utilizando un conjunto de funciones desarrolladas en MATLAB.

Desde el punto de vista de la implementación *hardware*, decir que el módulo de ethernet trabaja con dos dominios de reloj, uno impuesto por el dispositivo externo que resuelve el enlace físico de ethernet y el otro es el propio de los datos que se desean transmitir y recibir. Internamente trabaja con FIFOs de doble reloj donde se almacenan las tramas de datos que se reciben/envían desde/hacia el ordenador. Cuando se reciben comandos, este bloque se encarga

de procesarlos y generar las señales de control y escritura de registros de los módulos descritos anteriormente.

Adicionalmente, se dispone de algunas configuraciones del AWG que se realizan mediante botones e interruptores y es este módulo el encargado de eliminar rebotes y sincronizarlas con el dominio de reloj de destino.

4.1.4 Puesta en marcha

Para la implementación de un diseño de esta complejidad es necesario dotar al desarrollo de ciertas herramientas de depuración de errores y realizar algunas tareas de calibración de los DACs. Para depurar errores se ha añadido la capacidad de poder leer el contenido de las memorias y que el mismo sea enviado al ordenador mediante la conexión de gigabit ethernet. También se incorporó un módulo ILA (*Integrated Logic Analyzer*) para verificar la integridad de los datos en puntos críticos del sistema, en la versión final de este desarrollo se ha quitado este módulo.

Como se ha mencionado en el 4.1.1, el conversor DA dispone de 4 fases de reloj para muestrear los datos provenientes de la FPGA. Una vez que se ha comprobado, mediante el uso del analizador lógico integrado en la FPGA, que las muestras salen correctamente hacia el conversor, queda determinar cuál es la fase correcta para una determinada frecuencia de muestreo. Esto se logra a base de prueba y error utilizando un banco de pruebas como el que se muestra en la Figura 4.3.

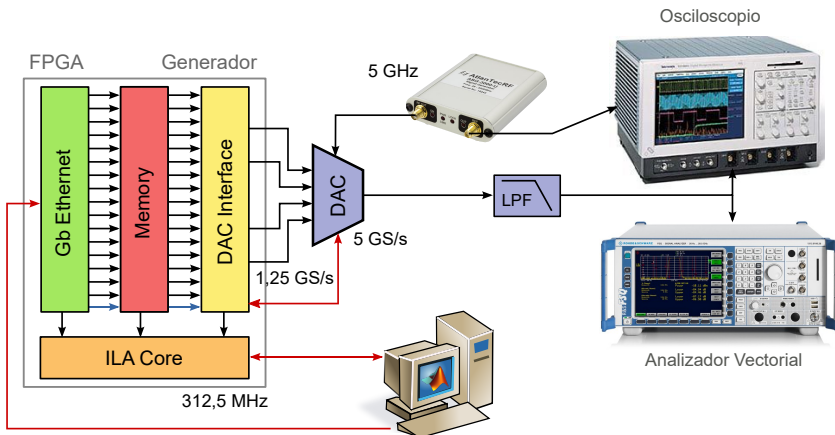
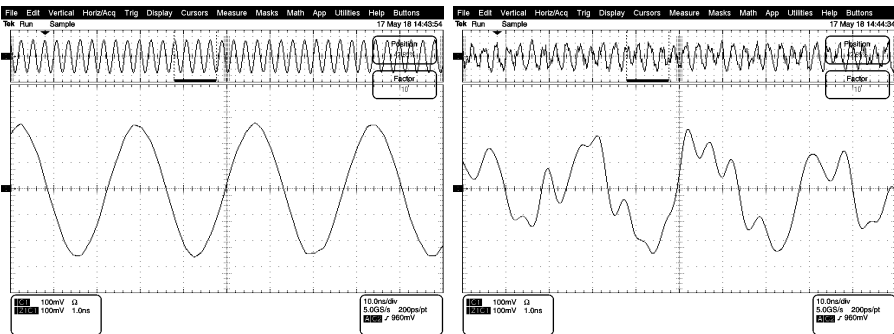


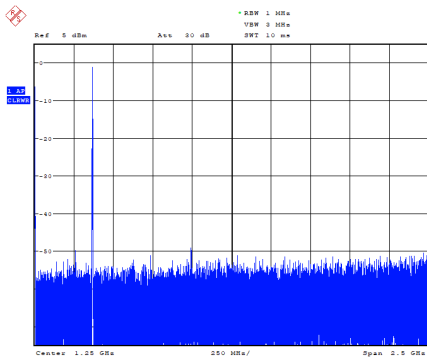
Figura 4.3: Banco de pruebas para verificación del control y generación del DAC.

Inicialmente se hicieron pruebas con señales de bajo contenido espectral que permitieran, mediante una simple inspección del osciloscopio o analizador de señal, determinar qué fase del reloj es la más adecuada para muestrear las señales diferenciales. Realizando este tipo de pruebas se llegó a la conclusión de que a simple vista (ver Figura 4.4) podría haber más de una fase válida, razón por la cual, es necesario realizar algún tipo de medición que permita determinar con más exactitud cuál es la fase adecuada. Para llevar adelante esta tarea se recurrió a generar señales OFDM, capturarlas con el osciloscopio y procesarlas en MATLAB para obtener los valores de EVM y BER. Analizando estos resultados se ha podido determinar la fase adecuada para cada uno de los conversores a diferentes frecuencias de muestreo.

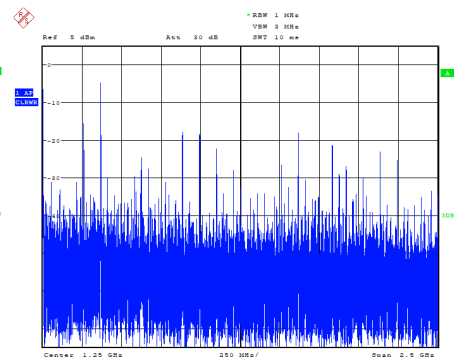


(a) Forma de onda para fase 0, 2 y 3

(b) Forma de onda para fase 1



(c) Espectro para fase 0, 2 y 3



(d) Espectro para fase 1

Figura 4.4: Pruebas de selección de fase del conversor DA con un tono puro de 366 MHz.

4.2 Receptor OFDM

El receptor OFDM que se describe a continuación funciona en tiempo real utilizando un convertor AD de 10 bits a una tasa máxima de 5 GS/s y un dispositivo FPGA Virtex-7. Todo el procesamiento digital de la señal se realiza dentro de la FPGA de forma paralela y en tiempo real, mientras que las mediciones de BER y EVM se obtienen mediante un post procesamiento *off-line* en el ordenador utilizando MATLAB.

El convertor AD muestrea la señal eléctrica de banda base y envía las muestras a un primer bloque dentro de la FPGA, denominado interfaz ADC, que se encarga de procesar las muestras para adecuarlas a la interfaz de entrada del bloque de DSP, además de realizar tareas de configuración y calibración del propio convertor. Una vez obtenidos los bits de información, estos se almacenan en bloques de memoria internos de la FPGA y posteriormente son enviados al ordenador mediante la interfaz de gigabit ethernet, de estas tareas se ocupa el módulo de control e interfaz de usuario (CUI).

4.2.1 Interfaz ADC

El convertor utilizado en esta tesis es el EV10AQ190A de la empresa E2V y cuyo diagrama de bloques se muestra en la Figura 4.5. Este convertor está compuesto de 4 ADCs de 10 bits que pueden alcanzar una frecuencia de muestro de 1,25 GS/s cada uno y pueden trabajar de 3 maneras diferentes: independientes (1,25 GS/s), de a parejas (2,5 GS/s) o entrelazados (5 GS/s). Al ser 4 convertidores (canales) independientes es necesario realizar una calibración analógica y digital, la primera consiste en igualar su comportamiento analógico mediante un ajuste de ganancia, *offset* y fase, y la segunda en muestrear correctamente los 40 pares diferenciales y colocarlos en un mismo dominio de reloj.

La calibración analógica se ha descrito en el capítulo 2. Para poder realizar dicha calibración es necesario acceder a los registros internos del convertor vía SPI (*Serial Peripheral Interface*). Este módulo junto con el CUI y algunas funciones realizadas en MATLAB son los que posibilitan la lectura/escritura de estos registros y facilitan la calibración del ADC desde el ordenador.

Cada uno de los 4 convertidores que integran el EV10AQ190A transmiten sus datos (10 bits) mediante 10 pares diferenciales síncronos con un reloj propio, es decir que el ADC tiene como salida de datos 40 bits y 4 relojes. Debido a las diferencias de caminos que pueden existir entre los pines de salida del ADC y los de entrada a la FPGA, se hace necesario implementar una etapa

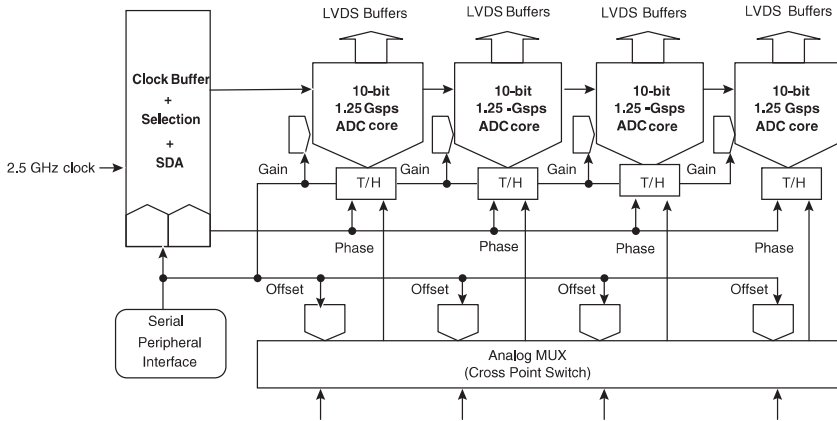


Figura 4.5: Diagrama de bloques del convertor EV10AQ190A.

de calibración digital que compense las diferencias de caminos y que coloque a las muestras en el mismo dominio de reloj.

En la Figura 4.6 se muestra el diagrama de bloques de la interfaz ADC, donde en primer lugar se encuentran una serie de búferes diferenciales de entrada (IBUFDS), a continuación se introducen retardos analógicos (IDELAY) a todos los bits recibidos y, por último, unos dispositivos dedicados a reducir la tasa de las muestras (ISERDES) de 1,25 GS/s a 312,5 MS/s. Después de esta primera etapa nos encontramos con 160 bits (16 muestras) y 4 relojes de 312,5 MHz, pudiendo darse el caso que se produzcan violaciones de tiempo (*setup/hold*) para algunos bits, que el retardo entre los bits de una misma muestra sea mayor a un ciclo de reloj y que la diferencia de tiempo entre muestras de los diferentes convertidores supere un ciclo de reloj. Es por todo esto que, a continuación, 3 subsistemas realicen las tareas de alineamiento que se detallan en los siguientes apartados.

Los bloques de la izquierda de la Figura 4.6 (bloques amarillos) pertenecen a los recursos dedicados de entrada/salida de la FPGA, aquí no se utilizan descripciones de comportamiento, sino que se ha de instanciar los componentes específicos del dispositivo utilizado con su correspondiente configuración. Para el camino de datos de los 4 canales se han utilizado: 40 búferes diferenciales (IBUFDS), 40 retardos analógicos variables (IDELAY) y 40 convertidores serie/paralelo (ISERDES) trabajando en modo DDR. Para el camino de reloj de los 4 canales se han utilizado: 4 búferes diferenciales (IBUFDS), 4 retardos analógicos fijos (IDELAY) y 8 búferes de reloj (BUFIO y BUFR).

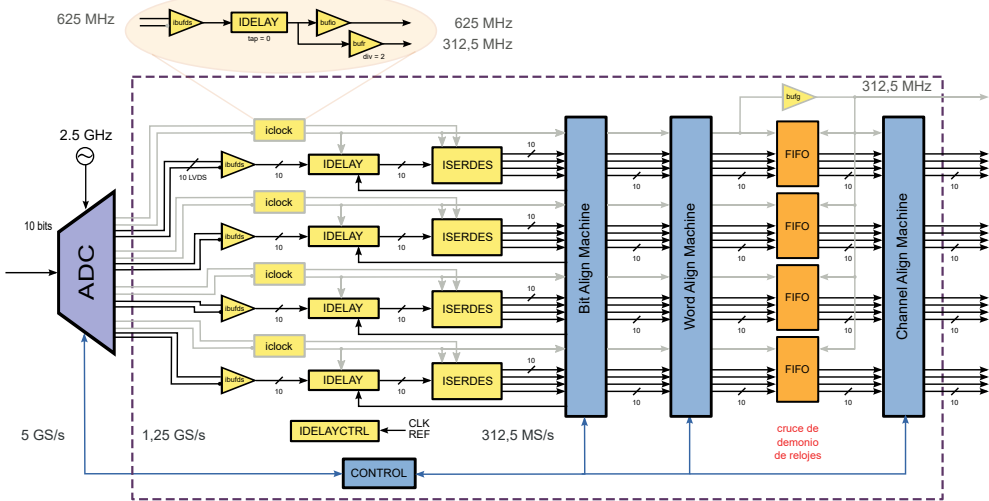


Figura 4.6: Diagrama de bloques de la interfaz con el ADC.

Las redes BUFIO/BUFR maximizan el rendimiento del reloj a expensas de algún inconveniente. BUFIO y BUFR son relojes regionales que no pueden abarcar todo el chip como un reloj global, pero los relojes globales tienen un rendimiento menor que los regionales [131]. Como los restantes módulos del receptor deben trabajar con un reloj global, la transferencia de datos entre los dominios regionales y globales se debe hacer mediante una FIFO de reloj dual.

4.2.1.1 Alineamiento de bits

Este subsistema es el encargado de que cada uno de los 10 bits esté correctamente muestreado tomando como referencia el reloj al cual pertenece la muestra. Para ello se deben modificar las líneas de retardo de cada uno de los 10 bits que conforman la muestra para encontrar el instante óptimo de muestreo de cada bit. Esto se logra colocando el flanco del reloj en el centro del diagrama de ojo de cada bit mediante el incremento de los retardos analógicos introducidos en el camino de datos.

Para realizar dicha tarea se configura al ADC vía SPI en modo intermitente (*Flashing mode*), lo que implica que el ADC presenta en sus 40 bits de salida un 1 lógico cada 11 ciclos de reloj, el resto del tiempo las salidas permanecen a 0 lógico. De esta manera se genera en todas las salidas un patrón conocido totalmente independiente de las entradas analógicas del conversor. Ahora bien,

dentro de la FPGA sólo se puede afectar los retardos de 10 bits de cada canal a pesar de trabajar con 4 muestras por canal, lo cual no es un inconveniente ya que los retardos aplicados afectarían por igual a las 4 muestras. Basta entonces con tomar los 10 bits de una de las 4 muestras del mismo canal y comenzar a variar los retardos introducidos mediante los IDELAYs hasta obtener el instante óptimo de muestreo.

La manera de hacerlo es mediante una máquina de estados dentro de la FPGA que analiza el diagrama de ojos de cada bit tal como se muestra en la Figura 4.7. En cada paso de este algoritmo se puede incrementar o disminuir el retardo de la ruta de datos. Este algoritmo mide el ancho del diagrama de ojo en términos de pasos de retardos (75 ps para Virtex-7) y luego lo centra con respecto al flanco de subida del reloj del canal. Los pasos 1 y 2 están dedicados a modificar el retardo hasta encontrar la transición entre el 1 y el 0 lógico, mientras que los pasos 3 y 4 están dedicados a medir el ancho del ojo. El paso 5 se encarga de partir esta diferencia de incrementos realizados y centrar así el flanco del reloj con en el centro del ojo.

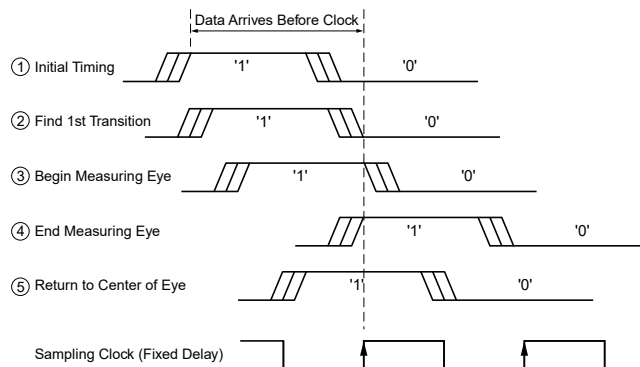


Figura 4.7: Algoritmo de alineamiento de bit de la interfaz del ADC.

El *jitter* se representa en la Figura 4.7 como líneas de transición múltiples. Tomar en consideración la existencia del *jitter* es una parte crucial del algoritmo. En el paso 2, se encuentra la primera transición de 0 a 1. Si no hay *jitter*, entonces la medición del ancho del ojo puede comenzar instantáneamente después de encontrar dicha transición. Sin embargo, cuando se considera el *jitter*, el algoritmo debe encontrar la primera transición, moverse a través de esta y luego comenzar a buscar la segunda transición. Si el algoritmo no se mueve a través de la primera transición de forma inteligente, corre el riesgo de detectar erróneamente la segunda transición.

4.2.1.2 Alineamiento de palabras

Con el ADC en modo intermitente (10 ceros y 1 uno), se escoge uno de los bits de una muestra del canal como referencia y se modifican (de ser necesario) el retardo de los restantes bits de las muestras del canal para que estén todos alineados con el bit de referencia. Esta vez los retardos se generan con registros de desplazamiento ya que los bits ya se encuentran alineados con el reloj.

Dentro de la FPGA se busca la presencia de un 1 lógico en cada uno de los 40 bits del mismo canal y se calcula el retardo en número de ciclos con respecto al bit de referencia. Una vez obtenidos estos valores y aplicados a los registros de desplazamiento, se realizan dos comprobaciones: una dentro de la misma FPGA y otro en el ordenador. Para la primera se configura el ADC en modo intermitente y se controla que el patrón capturado en todos los bits del canal al cabo de 1, 2, 3 y 4 segundos sea el mismo, de manera tal de descartar cualquier variación debido a la dispersión de parámetros. Para la segunda comprobación se configura el ADC en modo rampa (desde 0 hasta 1023) y se capturan 16384 muestras por canal y se controla que cada muestra de cada canal sea igual a la anterior muestra más 1 (suma módulo 1024).

4.2.1.3 Alineamiento de canales

Hasta ahora se ha trabajado con los 4 canales del ADC de manera independiente y se ha logrado obtener el instante de muestreo óptimo para cada bit y una alineación temporal de los mismos para obtener en un mismo instante de tiempo todos los bits correspondientes a la misma muestra del canal.

Con el ADC configurado en modo rampa se busca en cuál de los 4 canales está presente el menor valor y se calcula la diferencia de valores que existen con las muestras de los otros 3 canales. Al ser una rampa esa diferencia de valores se puede traducir en ciclos de reloj, y las diferencias entre los canales se puede corregir mediante registros de desplazamiento que apliquen retardos equivalentes a los valores obtenidos. Este alineamiento se realiza de manera automática dentro de la FPGA. Es necesario que todas las muestras se encuentren en el mismo dominio de reloj para poder alinear los canales, esto se logra utilizando 4 FIFOs de reloj dual.

4.2.1.4 Puesta en marcha

Para implementar un interfaz que pueda controlar y configurar un convertor de esta complejidad es necesario incluir herramientas de depuración de errores y poder acceder desde el ordenador tanto a los registros internos del ADC como a las muestras capturadas. Para realizar la calibración analógica y la digital, se ha añadido una memoria de almacenamiento y la posibilidad de que el contenido de las mismas sea enviado al ordenador mediante la conexión de gigabit ethernet para ser analizado desde MATLAB. También se incorporó un módulo ILA (*Integrated Logic Analyzer*) para verificar la integridad de los datos en puntos críticos del sistema, en la versión final se eliminado.

En el capítulo 2 se ha explicado la calibración analógica, la cual hace necesaria un banco de pruebas como el que se muestra en la Figura 4.8, donde se pueden generar señales de prueba mediante un generador de señales que se encuentra sincronizado con el reloj del convertor, capturar las muestras, enviarlas al ordenador y configurar al convertor vía SPI.

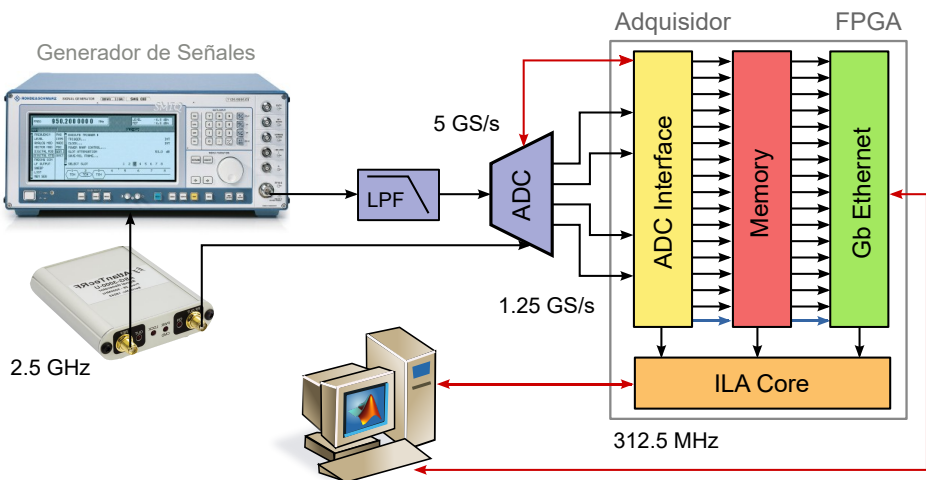


Figura 4.8: Banco de pruebas para verificación del control y adquisición del ADC.

Aunque la calibración digital es un poco más automática y se realiza en su totalidad dentro de la FPGA, requiere acceder a configurar el ADC en los distintos modos de evaluación según la alineación que se esté realizando. El inicio de cada alineación también se le indica a la FPGA desde el ordenador mediante una serie de registros, así como también se informa del resultado de las mismas.

Este banco de pruebas es el punto de partida para la implementación del receptor OFDM. Incorporando los bloques de DSP, que a continuación se detallarán, antes de la memoria de almacenamiento y una lógica de control adicional, se puede obtener el receptor OFDM.

4.2.2 Sincronizador Temporal

En el apartado 3.2.1.2 se ha presentado el algoritmo de sincronización de tiempo paralelo y en apartado 3.2.1.3 su modelo de precisión finita. Este bloque no sólo detecta la presencia de la secuencia de entrenamiento, sino que una vez sincronizado, reordena las muestras para que puedan ser procesadas por los bloques que se encuentran a continuación (ver Figura 3.10). Este bloque cuenta con 16 entradas de datos (de 10 bits cada una), 1 entrada de aviso de final de trama (EOF), 1 salida de datos válidos (ODV), 1 salida de detección de sincronismo válida (OSV) y 16 salidas de datos (de 10 bits cada una). El aviso de final de trama viene de una máquina de control que le indica al TSA que comience a buscar la presencia de la TS, para que una vez detectada, abandone la búsqueda y, de esta manera, evite falsas detecciones. Una vez determinado el último pico de la TS, se procede a desplazar temporalmente las muestras de forma tal que la salida de datos 0 coincida con el comienzo del intervalo de guarda (ver Figura 3.12) y se activan las señales ODV y OSV. La primera de ellas da el aviso al siguiente bloque para que comience a procesar las muestras, mientras que la segunda, le indica a la máquina de control que el sincronismo ha sido detectado.

La implementación sigue un esquema muy similar al presentado en la Figura 3.15, donde en primer lugar se encuentra un bloque de almacenamiento y cuantificación (1 bit) de muestras actuales y pasadas necesarias para realizar la correlación cruzada. A continuación 16 bloques idénticos que implementan la correlación cruzada de una secuencia de 32 muestras mediante un sumador en árbol implementado mediante LUTs (*LookUp Tables*) de 5 niveles segmentados para reducir el camino crítico. Luego 16 bloques idénticos que implementan el promedio utilizando un sumador (LUTs) y tres registros de 6 bits cada uno. Para determinar el pico se colocan 16 comparadores de 6 bits y las salidas de los mismos se conectan a un decodificador con prioridad para que determine cuál es la última muestra del pico, ya que podría haber más de una muestra que supere el umbral (depende de la anchura y ruido del pico). Por último, mediante registros de desplazamiento y multiplexores se realiza el alineamiento de los datos de salida con el comienzo del GI.

Todo el diseño se encuentra desarrollado en aritmética de punto fijo, segmentado y paralelizado para alcanzar el *throughput* requerido con un 100 % *HUE*. La arquitectura TSA desarrollada en este trabajo ha sido implementada en un dispositivo FPGA Virtex-7 XC7VX485T-2 y el número de registros y LUTs utilizados es 1.481 y 2.180, respectivamente. Se obtuvo una frecuencia máxima de operación de 625 MHz superando los 312,5 MHz necesarios para trabajar a 5 GS/s, lo que permitiría trabajar a una velocidad de muestreo de hasta 10 GS/s.

4.2.3 Transformada Rápida de Fourier (FFT)

Este procesador ha sido diseñado para ser implementado en diferentes dispositivos FPGA de Xilinx y obtener el mejor rendimiento posible, por lo que es un diseño parametrizado, con una segmentación muy cuidada y un gran control del *fanout* de las señales de control, de modo que la frecuencia de trabajo máxima del diseño es muy cercana a la máxima frecuencia de conmutación del elemento lógico más lento dentro del FPGA.

El módulo FFT utiliza aritmética de punto fijo y su descripción *hardware* se ha realizado utilizando la herramienta de alto nivel System Generator, desarrollada por Xilinx para trabajar en Simulink en el diseño e implementación de sistemas de procesamiento digital de la señal. Los módulos de reordenamiento de muestras de entrada y salida han sido descritos en lenguaje Verilog. La longitud de FFT se puede configurar en tiempo de ejecución y puede tomar los siguientes valores: 16, 64, 256 y 1024. Todos los módulos se han verificado utilizando un modelo de precisión finita realizado en MATLAB.

4.2.3.1 Elemento Computacional

Este bloque es el núcleo de la arquitectura PFFT y el único que realiza operaciones aritméticas: sumas, restas y multiplicaciones de números complejos. El CE se repite en cada etapa de la arquitectura de manera sistólica. Debido a esto, el diseño y la implementación del elemento computacional es crítico para las prestaciones del procesador FFT. En el apartado 3.2.2.1.1 se han presentado las ecuaciones que describen al CE y en el apartado 3.2.2.4 se ha detallado el modelo de precisión finita. A continuación se verán los aspectos relacionados con la implementación: recursos utilizados, latencia y caminos críticos.

Si se toma el caso de una FFT de 1024 puntos ($N=1024$) y de 16 muestras en paralelo ($P=16$), y se utilizan estos valores en la ecuación 3.16, se obtiene que son necesarios 48 multiplicadores y 160 sumadores de números complejos.

Esto equivale a 192 ($48 \cdot 4$) multiplicadores y 416 ($48 \cdot 2 + 160 \cdot 2$) sumadores de números reales.

Para implementar los sumadores de números complejos y las multiplicaciones triviales del CE (lado izquierdo de la Figura 3.20) se emplean 16 sumadores de números reales (basados en LUTs) organizados en dos etapas de 8 sumadores cada una. Los tamaños de palabra de estos sumadores van desde los 10 bits hasta los 20 bits para el sistema OFDM desarrollado en esta tesis. Como es sabido, los sumadores de lógica distribuida pueden presentar caminos críticos en el diseño de sistemas gran coste computacional. Para mitigar esta posible limitación de la frecuencia máxima de operación se han introducido registros a la entrada y salida de cada uno de los sumadores para segmentar estas dos etapas, por lo tanto, la latencia acumulada es de 3 ciclos de reloj.

Los multiplicadores suelen ser un punto crítico de las implementaciones en dispositivos FPGA si no se utilizan los multiplicadores *hardware* (p. ej. DSP48) disponibles en el dispositivo. Como el receptor OFDM requiere de muchos multiplicadores y en particular este módulo representa casi la mitad del total, es por ello por lo que se ha optado por utilizar en todos los casos multiplicadores *hardware*. Para implementar los multiplicadores de números complejos del CE (lado derecho de la Figura 3.20) se ha decidido utilizar la opción que requiere 4 multiplicadores y 2 sumadores de números reales (ver ecuación 3.14) ya que puede ser implementado de manera compacta con 4 bloques DSP48 y sin ninguna otra lógica adicional, tal como se muestra en la Figura 4.9. La forma alternativa (ver ecuación 3.15) requiere un multiplicador menos pero 3 sumadores más, lo cual se puede implementar mediante 3 bloques DSP48s y lógica adicional, y no suele conseguir las mejores prestaciones. Para que los bloques DSP48 alcancen su máxima frecuencia de conmutación es necesario utilizar toda la segmentación interna, que en este caso es de 4 niveles. Por lo tanto, la latencia acumulada es de 7 ciclos de reloj.

Los factores de rotación (módulo igual a 1 y fase variable) se calculan previamente y se almacenan en pequeñas memorias implementadas con lógica distribuida y sus salidas se registran para compensar el retardo en la ruta de datos, de esta manera se logra disminuir la posibilidad de una ruta crítica y dar mayor flexibilidad a la herramienta de emplazamiento. Estos factores son coeficientes trigonométricos constantes que multiplican los datos modificando su fase, pero no su amplitud. Estos factores de rotación tienen un ancho de palabra de 18 bits, de los cuales se destinan 1 bit para la parte entera y 17 bits para la parte fraccionaria.

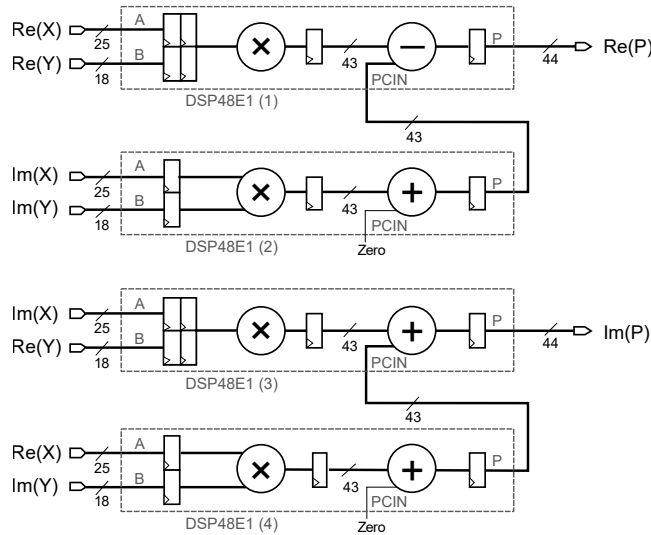


Figura 4.9: Multiplicador de números complejos, X corresponde a valores intermedios de la FFT e Y corresponde a valores de factores de rotación.

Si se utilizara redondeo a la salida del sumador y restador interno del multiplicador complejo (ver detalle en la Figura 3.20), se necesitarían 6 sumadores más por cada CE. Esto aumentaría los recursos de *hardware* utilizados y la latencia del módulo. Sin embargo, si se usa el truncamiento, se cumple el límite HD-FEC (ver Figura 3.25) y el coste computacional no se incrementa. Por este motivo, se decide utilizar truncamiento a la salida del CE.

Los bloques DSP48 no se encuentran distribuidos por toda la FPGA, sino que están agrupados en ciertas columnas del dispositivo. La falta de flexibilidad para emplazar estos componentes puede ocasionar caminos críticos entre los elementos que le preceden o suceden. Es por esto que se colocaron registros extras a la entrada y salida de los multiplicadores complejos, lo cual eleva la latencia del CE a 9 ciclos de reloj. La única excepción es para los CEs de última etapa, que no tiene multiplicadores complejos y cuya latencia total es de 3 ciclos de reloj.

4.2.3.2 Conmutador de demoras

El conmutador de demoras se puede dividir por en tres etapas, donde la primera y la última se corresponden con los retardos introducidos sobre los caminos de datos a la entrada y salida del módulo. La etapa del medio se corresponde con el conmutador de muestras que en su interior está integrado por 4 multiplexores 4:1 de varios bits. Los elementos de retardo se han implementado mediante registros de desplazamiento basado en LUT (*Shift Register LUT - SRL*) seguidos de un flip-flop (FF) para garantizar la segmentación. La primera entrada y la última salida no poseen elementos de demora (ver Figura 3.21) y podrían dar origen a un camino crítico o dificultar la tarea de emplazamiento, por ello, se han registrado todas las entradas y salidas. La señal de control del DC se conecta a la entrada de selección de 4 multiplexores 4:1 de $a + 2k$ bits cada uno, lo que implica un *fan-out* de $4a + 8k$. Este gran *fan-out* se reduce utilizando un árbol de registro equilibrado.

Los DCs de la arquitectura PPFFT tienen valores internos de retardos (ver Tabla 3.3) y una secuencia de conmutación determinada por la etapa del algoritmo a la cual pertenecen. Estas señales de conmutación se generan a partir de la selección de 2 bits de un contador de 6 bits que funciona con una frecuencia de reloj igual a la frecuencia de muestreo dividida por 16. Por ejemplo, para los DCs de la etapa 3 ($k=3$ y $f=4$), los bits 2 y 3 del contador son utilizados como señales de conmutación (ver Figura 3.22).

4.2.3.3 Reordenamiento de entradas y salidas

Los reordenamientos de las muestras de entrada y salida son diferentes y dependen de dos variables: la longitud de FFT (N) y del número de muestras en paralelo (P), tal como se ha mencionado en el apartado 3.2.2.3. Además, se debe tener en cuenta que la longitud de la FFT se puede modificar en tiempo real. Solo algunos trabajos en la literatura [111, 115] proponen una solución para estos dos módulos, teniendo en cuenta una longitud variable y el proceso continuo de P muestras en paralelo; ambos orientados a diseños VLSI. Este nivel de complejidad obligó a diseñar una solución generalizada y configurable que permitiese resolver todos los escenarios de esta tesis, y porque no también, escenarios futuros.

La solución propuesta se basa en una matriz bidimensional de $P \times N/P$ para cada salida del módulo de reordenamiento de datos (P salidas), donde las filas representan las P entradas que están disponibles en el mismo ciclo de reloj (dimensión espacial) y las columnas representan esas entradas en diferentes

ciclos de reloj (dimensión de tiempo). Es decir, que se necesitan P matrices bidimensionales para realizar el reordenamiento de entrada o salida, lo cual implica una capacidad de almacenamiento total de $P \times N$ muestras.

Cada matriz bidimensional almacena las N muestras que ingresan a la PPFFT, tal como se muestra en Tabla 3.4, o que egresan de la PPFFT, tal como se obtendrían después de aplicar la transformada *digital-reversal* a la Tabla 3.4. En el apartado 3.2.2.3 se ha mencionado que es necesario realizar permutaciones en ambas dimensiones (espacial y temporal) para obtener las muestras en el orden deseado. Además, hay casos (p. ej. 256 o 1024-FFT) donde cada salida depende de cualquiera de las 16 entradas, como puede verse en Tabla 3.5 (p. ej. para $N=1024$ y $P=16$, $L=64$ y $M=256$). Por esta razón, esta matriz bidimensional ($16 \times N/16$) tiene que ser replicada 16 veces. Por otro lado, para evitar la pérdida de muestras debido a la sobrescritura (las salidas no se pueden generar hasta que todas las N muestras ya estén almacenadas), se emplea la técnica de doble búfer o *ping-pong buffering*, que obliga a duplicar la profundidad (columnas) de la matriz bidimensional. Por lo tanto, la capacidad de almacenamiento necesaria para llevar adelante esta solución de reordenamiento de muestras de entrada/salida es de $P \times 2N$ muestras.

Para implementar la matriz bidimensional, se han utilizado bloques de memoria RAM (BRAMs) en modo doble puerto simple (*Simple Dual-Port* - SDP) con un ancho de $16 \cdot d$ bits y una profundidad de $2N/16$, donde d es el ancho de palabra de las muestras a reordenar. Por lo tanto, para acceder a una de las N muestras almacenadas en la matriz bidimensional, es necesario indicar una dirección de lectura (columna) y seleccionar la fila adecuada (una muestra de d -bit) con un multiplexor de 16 a 1. La generación de la dirección de lectura de la RAM y la selección de entrada del multiplexor se realizan por medio de memorias ROM que tienen esta información almacenada para los diferentes valores de N y las P salidas del módulo. En consecuencia, se utilizan 4 ROM para las direcciones y 4 ROM para la selección de multiplexores, además, se utiliza un contador para generar la dirección de lectura. Como se dijo anteriormente, esta matriz bidimensional se replica 16 veces tal y como se muestra en la Figura 4.10. Este diseño aprovecha el uso de BRAMs que están disponibles en dispositivos FPGA.

Las salidas de los multiplexores y las entradas y salidas de las memorias han sido registradas para disminuir la aparición de caminos críticos y facilitar la tarea de emplazamiento (las BRAMs se encuentran en columnas específicas dentro de la FPGA). Cada matriz bidimensional está compuesta de varias BRAMs, por lo cual el *fan-in* de la entrada de dirección de lectura es superior a uno. La salida del contador, señal de control, se conecta a las 16 matrices

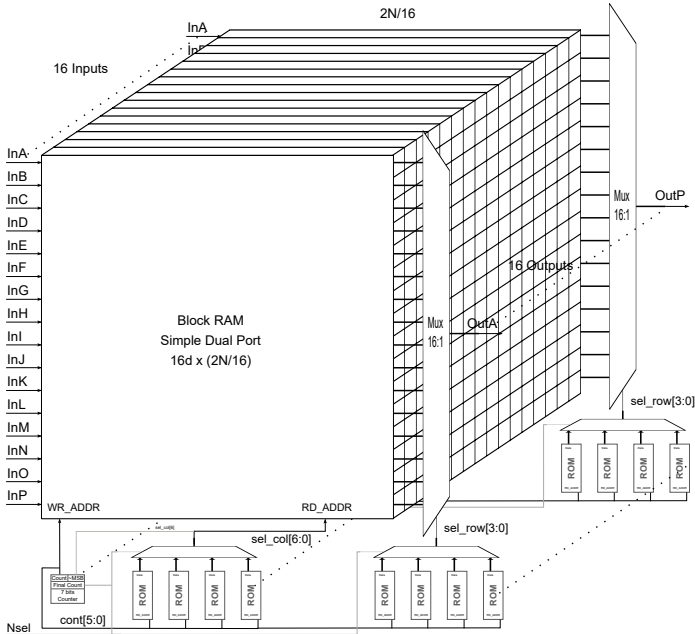


Figura 4.10: Implementación del hardware de reordenamiento de las muestras de entrada/salida.

bidimensionales, a los 32 multiplexores 4:1 y las 128 memorias ROM, con lo cual se hace necesario controlar el *fan-out* de esta señal mediante un árbol de registros. Por último, para garantizar la máxima frecuencia de trabajo de las BRAMs se utilizaron los registros internos que posee para sus entradas y salidas (*DI*, *RDADDR*, *WRADDR* y *DO*). De esta manera, el diseño tiene una latencia de 6 ciclos sobre el camino de datos. Para poder comenzar a leer las memorias hay que esperar a que se hayan escrito las N primeras muestras; lo cual requiere $N/16$ ciclos de reloj. En consecuencia, la latencia del módulo varía con el valor de N .

4.2.3.4 Resultados de Implementación

Todo el diseño se encuentra desarrollado en aritmética de punto fijo, segmentado y paralelizado para alcanzar el *throughput* requerido con un 100% *HUE*. La arquitectura PPFFT de longitud variable desarrollada en este trabajo, con 10 bits de ancho de palabra para las 16 entradas reales y 40 bits de ancho de palabra para las 16 salidas complejas, ha sido implementada en un dispositi-

vo FPGA Virtex-7 XC7VX485T-2 y los recursos utilizados son los siguientes: 14.017 LUTs, 31.578 registros, 100 BRAMs y 192 DSP48s. Las BRAMs sólo han sido utilizadas para implementar el reordenamiento de entradas y salidas, y los DSP48s para implementar los multiplicadores complejos de las *dragonflies*. Se obtuvo una frecuencia máxima de operación de 625 MHz superando los 312,5 MHz necesarios para trabajar a 5 GS/s, lo que permitiría trabajar a una velocidad de muestreo de hasta 10 GS/s.

4.2.4 Ecuador de canal

En el apartado 3.2.3 se han presentado los algoritmos que intervienen en la ecualización del canal y estos se pueden agrupar en dos etapas: estimación y compensación. En ese mismo apartado se ha realizado un análisis de precisión finita indispensable para reducir el coste *hardware* de este módulo. Una vez que el sincronizador de tiempo informa que ha detectado sincronismo el estimador de canal comienza a procesar las muestras de los LS y, una vez obtenida dicha estimación, se almacena en una memoria para que posteriormente, al llegar los símbolos OFDM al compensador, pueda ser leída y utilizada para compensar cada una de las portadoras.

El diagrama de bloques de la implementación del estimador de canal se presenta en la Figura 4.11 junto con los resultados del análisis de precisión finita. Para promediar los 4 LS se utilizan 16 sumadores y 16 registros de desplazamiento, el escalamiento final se realiza mediante una selección de bits (ver ecuación 3.18). El cálculo de la DFT del promedio de los LS se realiza mediante el módulo de FFT descrito en el 4.2.3.

A continuación se describe la implementación de la ecuación 3.23. Para calcular el módulo de LS_{rm} se utilizan 32 multiplicadores y 16 sumadores, los cuales se implementan con 32 DSP48s con segmentación interna, y para obtener la inversa de este valor se utilizan 16 memorias ROM de $2^{12} \times 18$ bits; todo esto corresponde al primer término de la ecuación. En la ecuación encontramos el producto de una variable que sólo puede valer ± 1 (LS_t) por el conjugado de un variable compleja. Esta operación únicamente afecta al signo de la parte real e imaginaria de la variable compleja (LS_{rm}) y, por lo tanto, se implementa utilizando un módulo de complemento a dos (ca2) y un multiplexor que es controlado por el valor BPSK de LS_t (almacenado en una pequeña memoria ROM) para la subportadora seleccionada. Para completar la implementación de la ecuación mencionada se utilizan 32 multiplicadores *hardware* con redondeo y segmentación interna.

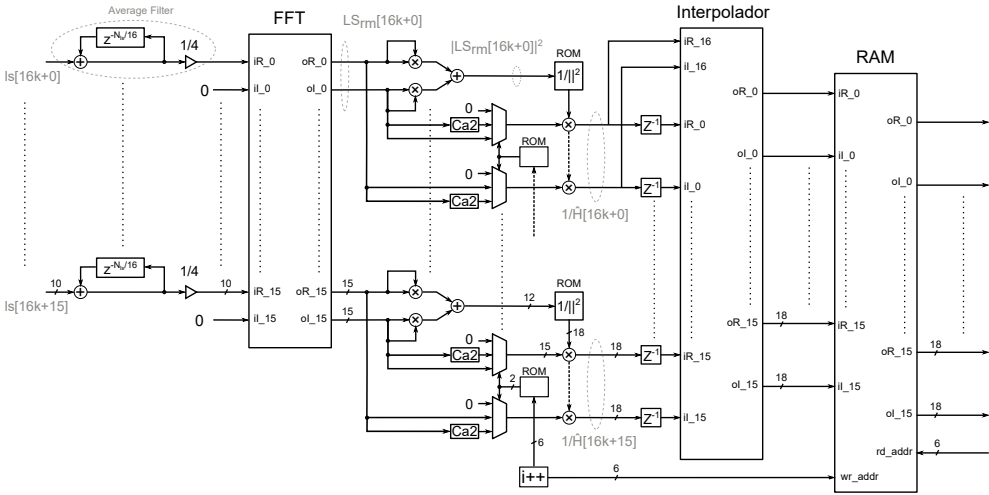


Figura 4.11: Diagrama de bloques del estimador de canal.

Por último, en el submódulo de estimación, se encuentran los bloques de interpolación y almacenamiento. El bloque de interpolación sólo actúa cuando el valor de N es 1024 y en caso contrario deja pasar las muestras. Esto es debido a que la interpolación utiliza dos subportadoras activas para obtener el valor de las tres subportadoras que se encuentran entre ellas, es necesario que este módulo tenga 17 entradas y 16 salidas. Para implementar la interpolación descrita en la ecuación 3.20 y evitar el uso de multiplicadores *hardware* se utilizan los denominados multiplicadores cableados (sumadores y rotaciones de bits). En la implementación de los sumadores se utiliza lógica distribuida y se han colocado 6 etapas de segmentación para reducir los caminos críticos y facilitar el emplazamiento. El bloque de almacenamiento se ha implementado mediante 16 memorias de bloque de $2^6 \times 36$ bits y se ha controlado el *fanout* de la señal de direcciones para mejorar la velocidad del diseño.

En lo que respecta a la etapa de compensación del canal no hay mucho por decir, lo único a destacar es que para implementar los multiplicadores complejos se utiliza la versión realizada con 4 multiplicaciones reales y 2 sumas reales, ya que con 4 DSP48s se puede implementar obteniendo una alta velocidad de funcionamiento. Para procesar 16 muestras en paralelo son necesarios 64 DSP48s. En la Figura 4.12 se muestra el diagrama de bloques de la implementación paralela del compensador de canal.

El escalamiento de la señal compleja proveniente del ecualizador se realiza mediante 32 multiplicadores embebidos con la siguiente configuración: 25 bits para la parte real o imaginaria de la señal, 18 bits para el factor de escala y el resultado se trunca sin redondeo a 6 bits. El factor de escalamiento se selecciona mediante un multiplexor que es controlado por el contenido de una memoria ROM en la que se encuentra almacenado el orden de modulación utilizado para cada una de las subportadoras.

Las entradas de datos del demodulador tienen un ancho de palabra 50 bits repartido en partes iguales para la parte real e imaginaria y el formato de cada una de ellas es s6.19. Para implementar todas las ecuaciones de desmapeo (3.24, 3.27, 3.25, 3.28, 3.26 y 3.29) hacen falta 62 comparadores reales (ver Tabla 3.6) que se implementan con centenas de LUTs y para reducir caminos críticos se han registrado las entradas y salidas.

La latencia del módulo de desmapeo es de 5 ciclos de reloj.

4.3 Resultados de implementación

A continuación se presentan los resultados de implementación del transmisor y receptor OFDM obtenidos para una FPGA Virtex-7 XC7VX485T-2. Para realizar la síntesis e implementación se ha utilizado la herramienta de software Vivado v2016.3 (64-bit) de Xilinx. Como estrategia global se ha optado por la que propone por defecto la herramienta Vivado, tanto para la síntesis como para la implementación. En la Tabla 4.1 y la Tabla 4.2 se encuentran detallados los principales recursos de *hardware* utilizados por los módulos que integran el transmisor y receptor, respectivamente. Ambos diseños han cumplido con el requerimiento de alcanzar una frecuencia de reloj de 312,5 MHz y han arrojado un *slack* positivo, lo que permitiría trabajar a más alta frecuencia de ser necesario.

Sobre los bloques DSP del receptor se destaca la utilización de recursos por parte del módulo FFT, que como era sabido, presenta la mayor carga computacional de todo el diseño; de ahí el esfuerzo en su diseño y optimización. La integración de todos los bloques desarrollados en esta tesis ha sido una tarea menor gracias a que de manera individual se han diseñado con un buen nivel de segmentación y un nivel de *fanout* de las principales señales de control acorde las necesidades de alta velocidad. En ambos diseños se observa que la interfaz de usuario utiliza una cantidad de recursos significativos y esto se debe exclusivamente a la complejidad de implementar un módulo de gigabit ether-

Tabla 4.1: Transmisor OOFDM - Recursos utilizados de la FPGA

Block	LUTs	FFs	Slices	BRAMs	DSP48s
DAC Interface	58	75	33	0	0
Storage & Control	1.659	11.380	6.780	768	0
User Interface	2.545	4.857	1.305	32	0
Total	4.262 (1.4%)	16.237 (2.57%)	8.118 (10.7%)	800 (77.67%)	0 (0.0%)

Tabla 4.2: Receptor OOFDM - Recursos utilizados de la FPGA

Block	LUTs	FFs	Slices	BRAMs	DSP48s
ADC Interface & Storage	8.686	8.911	4.343	232	0
TSA	1.481	2.180	532	0	0
Remove CP & Mean LS	2.415	3.823	1.087	0	0
FFT	14.017	31.578	7.086	100	192
Equalizer	4.198	9.213	2.008	24	128
Demapper	591	421	226	0	32
User Interface	2.545	4.857	1.305	32	0
Total	33.933 (11.2%)	60.983 (10.0%)	16.587 (21.8%)	388 (38%)	352 (13%)

net que trabaja con multitramas, con filtros de MAC/IP/PORT y que soporta comandos del protocolo ARP.

Tras una exhaustiva revisión bibliográfica en búsqueda de resultados de implementación para receptores OFDM ópticos con prestaciones y requisitos similares al desarrollado en esta tesis, sólo se han encontrado dos trabajos [60, 132] que satisfacen los requisitos mínimos para poder contrastar resultados de implementación. En la Tabla 4.3 se resumen las características de esos receptores y se muestran los recursos utilizados por estos diseños para un dispositivo FPGA Virtex-7 XC7VX485T-2.

En [60] se presentan los resultados de implementación de un receptor OFDM óptico que utiliza las mismas dos tarjetas de evaluación (ADC y FPGA) que las escogidas en esta tesis. Los autores han implementado un sincronizador temporal basado en la correlación cruzada, una FFT de 128 puntos en punto

Tabla 4.3: Comparación de características y recursos utilizados de la FPGA por receptores OOFDM

Ref.	Fs (MHz)	Fc (MHz)	P	LUTs	FFs	BRAMs	DSP48s
[60]	5000	156,25	32	69.937	66.562	410	426
[132]	2500	156,25	16	62.337	69.088	180	224
Propuesto	5000	312,5	16	33.933	60.983	388	352

fijo, un desmapeador M -QAM (256, 64 y 16), un estimador del SCFO basado en subportadoras pilotos y un analizador de tasa de errores. La compensación del SCFO se realiza a través de un oscilador controlado por tensión, lo cual encarece el sistema sin justificación alguna, ya que el SCFO obtenido no es lo suficientemente grande como para no ser compensado mediante algún algoritmo de DSP. La tasa de muestreo utilizada es de 5 GS/s y el grado de paralelismo es de 32, dando como resultado una frecuencia de trabajo de 156,25 MHz. De los 426 DSP48s utilizados, según detallan los autores, 236 son para la FFT, 124 para la compensación del canal, 62 para el desmapeo y 4 para la estimación del SCFO. Al trabajar con 32 muestras en paralelo es de esperar que todos los recursos *hardware* utilizados en [60] sean superiores a los aquí presentados. Al mismo tiempo, el utilizar una FFT de bastante menor longitud (128 en vez de 1024) reduce el coste *hardware* del sistema, ya que como se ha mencionado anteriormente, representa la parte más significativa de los recursos utilizados en un receptor OFDM. A su vez, al utilizar un mayor grado de paralelismo, la frecuencia de reloj disminuye y todo el diseño requiere de una menor segmentación, lo cual requiere de menos SRLs y menos FFs. Finalmente, el receptor presentado en [60] requiere de casi el doble de LUTs, un 20 % más de DSP48s y casi la misma cantidad de bloques de memoria y FFs.

En [132] se ha desarrollado un módem IM/DD OOFDM que utiliza una frecuencia de muestreo de 2,5 GS/s y unas pocas subportadoras (15 de baja frecuencia y 10 de alta frecuencia) con modulaciones muy bajas (16-QAM y QPSK) utilizando dispositivos Virtex-7. Los autores han implementado un sincronizador temporal y estimador de canal de baja complejidad, un estimador y compensador del SCFO, y una FFT de 128 puntos. En este caso los autores han decidido paralelizar por 16 el diseño, dando como resultado la misma frecuencia de trabajo que en [60] (156,25 MHz). Según detallan los autores, los 224 DSP48s se distribuyen de la siguiente manera: 124 para la FFT, 64 para el ecualizador de canal, 32 para el desmapeo y 4 para la estimación del SCFO. Al trabajar con 16 muestras en paralelo, una menor frecuencia de trabajo y

una FFT de inferior longitud, es de esperar que todos los recursos *hardware* utilizados en [132] sean inferiores a los aquí presentados. Además, los autores han considerado que para las primeras subportadoras el módulo de la estimación del canal se mantiene casi constante y que a las últimas subportadoras (QSPK) esto no le afecta. Por lo tanto, han optado por sólo compensar la fase del canal, lo cual les ha permitido reducir el coste *hardware* del receptor. Sin embargo, el receptor presentado en [132] requiere de casi el doble de LUTs, un 15 % más de FFs, un 37 % menos de DSP48s y un 54 % menos de bloques de memoria.

Ambos diseños utilizan una frecuencia de trabajo bastante inferior (la mitad) a la utilizada en esta tesis y ambos trabajos implementan una FFT de longitud fija (128 puntos). Lo primero implica trabajar bastante por debajo de la máxima frecuencia de operación del dispositivo FPGA, lo cual requiere de una menor segmentación y permite despreocuparse del *fanout* de las señales de control, además de acelerar la etapa de rediseño por la aparición de caminos críticos. Lo segundo, simplifica el diseño del procesador FFT, requiere de menos elementos computacionales (mariposas) y menos bloques de memoria (reordenamiento de entradas y salidas). Sin embargo, cabe resaltar que el número de LUTs y FFs utilizados no es menor al del receptor aquí presentado.

4.4 Simulación

Como se ha dicho durante todo el capítulo, se dispone de un modelo de simulación del sistema OFDM completo en MATLAB con posibilidad de configurar en punto fijo cada uno de los algoritmos y así obtener resultados y señales intermedias que permitan ser usadas como referencias en la simulación funcional del transmisor y receptor OFDM. Con esta simulación funcional no se busca evaluar el comportamiento de cada uno de los bloques sino el del conjunto. Cada bloque de la cadena de procesamiento fue verificado de manera independiente. Esta simulación funcional, junto con la no existencia de problemas de *timing* en la implementación, garantizan un correcto funcionamiento *hardware* y hacen innecesaria la simulación tras la implementación.

Tras la codificación del receptor, se desarrolló un banco de pruebas con capacidad de utilizar dos conjuntos de estímulos/resultados diferentes. El primero se corresponde al caso más ideal donde sólo se toma en cuenta la precisión finita del sistema y se anulan los efectos de todos los componentes que afectan a la señal OFDM. Estos estímulos y resultados se han generado con el simulador realizado en MATLAB. El segundo conjunto tiene que ver con señales reales

capturadas con el conversor AD y generadas con el transmisor OFDM descrito en el apartado 3.1 en distintos esquemas de medición (EBTB, OBTB, SSMF). Se ha realizado una comprobación bit a bit de los datos de salida del receptor para los diferentes esquemas y de esta manera se ha validado el correcto funcionamiento de la codificación del receptor.

Para verificar el funcionamiento de la etapa de control e interfaz de usuario de ambos diseños, se ha realizado en MATLAB un generador de tramas de UDP/ethernet que permite simular el envío de datos o comandos de configuración a ambos sistemas. De esta manera se ha podido simular el comportamiento completo del sistema sin tener la necesidad de recurrir al uso de módulos ILA para depurar errores, lo que implica una reducción del tiempo de depuración y la independencia que implica no tener que disponer del *hardware*.

Por último, decir que la verificación experimental en tiempo real del módem OFDM óptico se encuentra en el capítulo 5.

4.5 Conclusiones

Se ha desarrollado un módem OFDM óptico que trabaja en tiempo real a una tasa 5 GS/s utilizando dos tarjetas de evaluación que contienen un dispositivo FPGA Virtex-7, que se comunican a un ordenador mediante una interfaz gigabit ethernet y pueden ser controladas con total autonomía desde MATLAB, de manera tal que, se pueden realizar pruebas de larga duración con diferentes características.

Los diseños de la interfaz de ambos conversores y la puesta en marcha de estos no ha sido tarea fácil, la documentación y el material suministrado por los fabricantes es bastante escueto y muchas veces incompleto o incorrecto. Los elementos específicos de E/S de la FPGA tienen varias opciones de configuración y diversos modos de funcionamiento, sumado a que las notas de aplicación de Xilinx dedicadas a este tipo de interfaces de alta velocidad no se pueden aplicar a estos conversores, lo cual ha aumentado el grado de dificultad.

Los bancos de pruebas desarrollados para la puesta en marcha de los conversores (Figuras 4.3 y 4.8), no sólo han servido como punto de partida para el transmisor y receptor, sino que permiten generar y capturar cualquier tipo de señal eléctrica que no necesite una tasa de muestreo superior a 5 GS/s para posteriormente ser procesada en ordenador. Esta es una gran ventaja para poder estudiar y analizar otro tipo de modulaciones y nuevos algoritmos en entorno reales de comunicaciones ópticas.

El haber decidido trabajar con 16 muestras en paralelo, la eliminación de la dependencia entre salidas de cada uno de los algoritmos de DSP, el especial cuidado en la segmentación y el control del *fanout* de las principales señales de control, han dado un excelente resultado. Tanto el transmisor como el receptor pueden trabajar a mayor velocidad, lo cual es un objetivo buscado, ya que de aparecer conversores de mayor tasa de muestreo se podrán utilizar sin mayores cambios. La rápida integración de todos los bloques y la posibilidad de incorporar ,de manera sencilla, nuevos algoritmos de DSP, es resultado de las medidas adoptadas durante todo el diseño e implementación.

Por último, se han presentado los resultados implementación *hardware* del receptor OFDM sobre un dispositivo FPGA Virtex-7 y se han comparado con los obtenidos por receptores de similares características. Cabe destacar que los resultados obtenidos son mejores (se reducen los recursos *hardware*) que los publicados para el receptor con exactamente la misma tasa de muestreo.

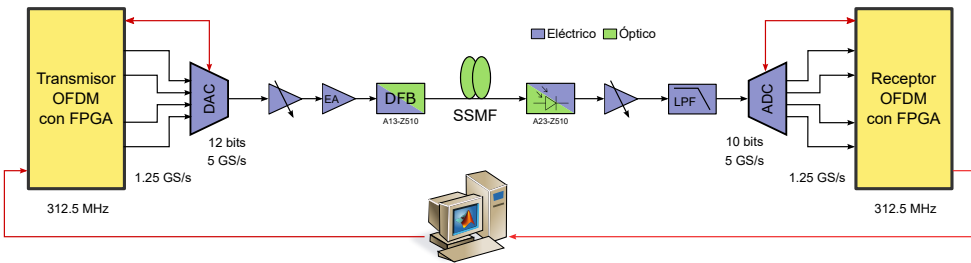
Plataforma de experimentación

En los capítulos anteriores se ha descrito cómo se han caracterizado todos los elementos que integran este sistema de comunicaciones ópticas, se han diseñado los algoritmos de procesamiento digital de la señal y se han implementado en dispositivos FPGAs. Ahora ha llegado el momento de poner todo a prueba, realizar los ajustes finales y medir las prestaciones del sistema para los diferentes escenarios.

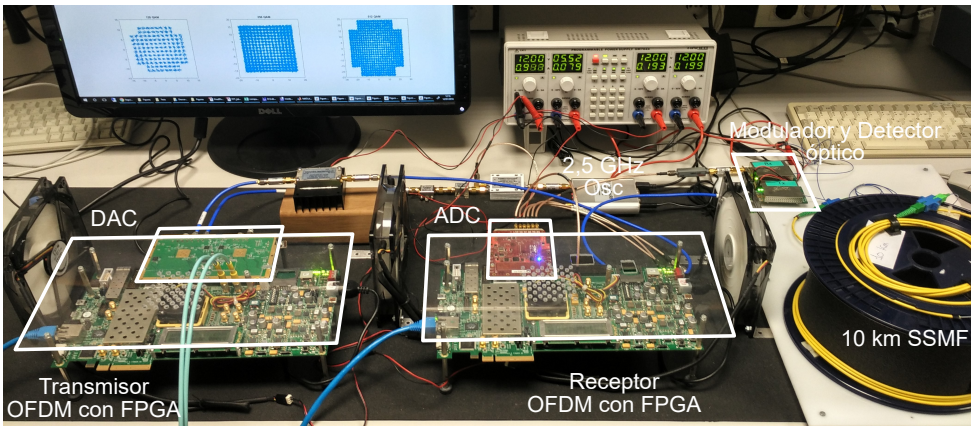
A continuación se describe la plataforma de experimentación, la estimación de canal obtenida para las diferentes configuraciones y los resultados de velocidad y eficiencia espectral para carga constante y variable. Después se presentan las características y prestaciones de otros sistemas que se pueden encontrar en la bibliografía de estos últimos años y se realiza un análisis comparativo contra el sistema desarrollado en esta tesis. Por último, se presenta un análisis cuantitativo de cómo afecta el SCFO a este sistema.

5.1 Descripción de la plataforma

Para realizar las mediciones que a continuación se presentan, se ha montado un banco de pruebas óptico de modulación y detección directa, de bajo coste y con procesamiento de las muestras en tiempo real, tal como se muestra en la Figura 5.1. Las muestras de la trama OFDM son generadas en el ordenador utilizando MATLAB y son enviadas al transmisor mediante una interfaz de gigabit ethernet. El transmisor está compuesto por dos tarjetas de evaluación: la VC707 de Xilinx que contiene una FPGA Virtex-7 y la FMC2657 de Evis que contiene dos DACs MD657B de 12 bits.



(a) Diagrama de bloques



(b) Experimento de laboratorio

Figura 5.1: Configuración experimental para el sistema IM/DD OOFDM en tiempo real.

La señal OFDM eléctrica es amplificada y atenuada (para ajustar el rango eléctrico) antes de la conversión electro-óptica. Un láser monomodo de realimentación distribuida (*Distributed Feedback Laser* - DFB) y modulación directa,

operando en una longitud de onda de 1550 nm, es modulado por la señal OFDM eléctrica amplificada y cuya potencia de salida óptica de +3 dBm. Sin utilizar un amplificador óptico, la señal se inyecta en fibra óptica monomodo (SSMF) y la señal OFDM de banda base recibida se obtiene a través de un fotodiodo (PD) InGaAs de alto rendimiento con un ancho de banda de 3,0 GHz.

El nivel de potencia de la señal de salida del fotodiodo se ajusta para mantener un nivel óptimo de valor de pico a pico en la entrada del ADC, de esta manera se aprovecha al máximo el rango dinámico del conversor y se evita la saturación. La señal fotodetectada se pasa a través de un filtro pasa bajo (LPF) con un ancho de banda de 2.343 MHz para evitar el solapamiento y se muestrea a 5 GS/s con un ADC EV10AQ190A de 10 bits. Este conversor está disponible en la tarjeta de evaluación FMC126 que se encuentra conectado a una tarjeta de evaluación VC707 de Xilinx que contiene una FPGA Virtex-7. Finalmente, las muestras capturadas se procesan en tiempo real dentro de la FPGA y los bits obtenidos se envían a un ordenador, a través de la interfaz gigabit ethernet, para calcular la BER y EVM de las subportadoras mediante MATLAB.

5.2 Estimación del canal

Una de las primeras mediciones realizadas con esta plataforma experimental está relacionada con la obtención de la estimación del canal y el EVM promedio por subportadora. Estas medidas se muestran en las Figuras 5.2 y 5.3 respectivamente, para lo cual se han cargado todas las subportadoras con una modulación 4-QAM y se han transmitido más de 400 símbolos OFDM con 512 subportadoras ($N=1024$). En la Figura 5.2 se puede ver que la incorporación de la etapa electro-óptica aumenta la pendiente de atenuación en relación al esquema EBTB, dando como resultado una pérdida total de potencia de 12 dB en las más altas frecuencias producto de la excesiva respuesta paso bajo del sistema (DAC, amplificador, láser, fotodetector, filtro antisolapamiento y ADC). Por otra parte, se observa que el aumento de la longitud de la fibra afecta levemente la pendiente para las más altas frecuencias tal como se ha descrito en el apartado 2.3 (ver Figura 2.6) producto de la dispersión cromática de la fibra y el efecto *chirp* del DML [39].

La fibra óptica utilizada en estos experimentos tiene unas pérdidas ópticas de 0,19 dB/km según ha especificado el fabricante, lo cual da una atenuación eléctrica total de 4/8/16 dB para las longitudes de 10/20/40 km, respectivamente. Aunque la potencia óptica recibida disminuya, la amplitud de la señal eléctrica a la entrada del ADC se mantiene constante gracias a la variación

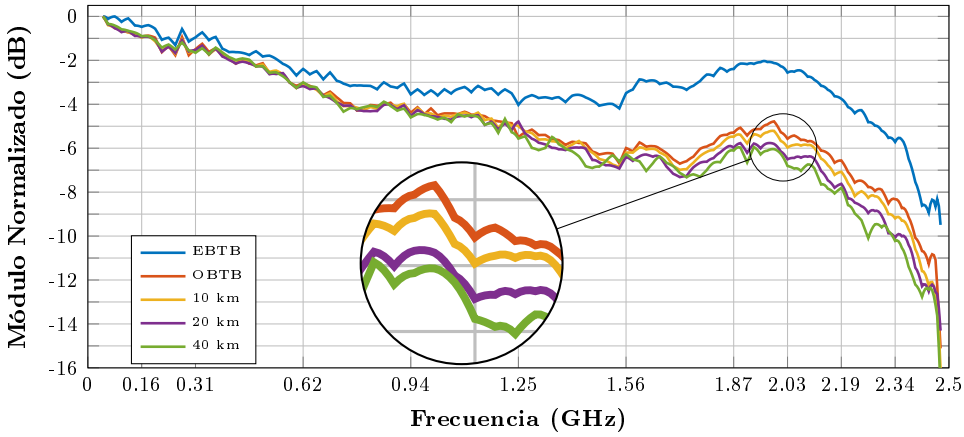


Figura 5.2: Estimación del canal para diferentes esquemas de transmisión.

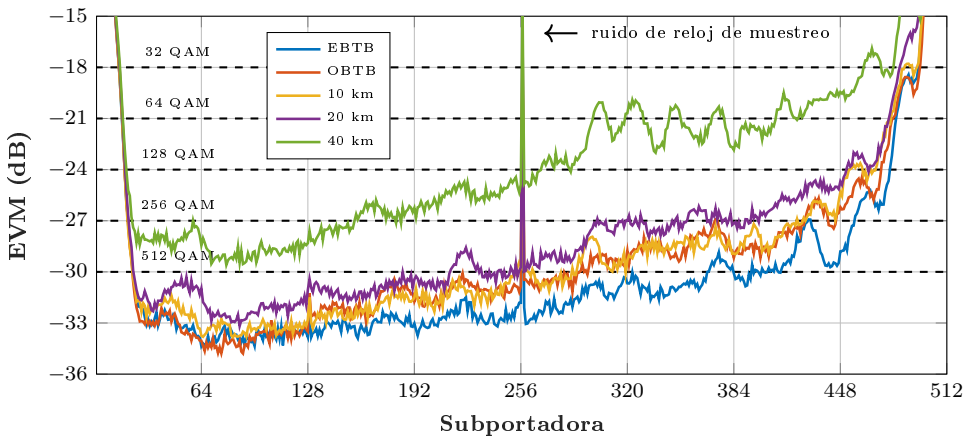


Figura 5.3: EVM por subportadora para diferentes esquemas de transmisión.

de la atenuación de la señal de entrada mediante diferentes atenuadores fijos que se conectan a la salida del fotodetector. Que los niveles de señal eléctrica recibida sean similares en todos los casos, hace que el ruido de cuantificación y otro efectos no deseados del convertor AD afectan por igual al sistema. Ahora bien, que la potencia óptica recibida varíe provoca que la SNR a la salida del fotodetector se degrade, por eso es importante utilizar un receptor óptico con amplificador de bajo ruido.

La manera de cuantificar cómo estos factores degradan la señal es realizando un promedio temporal del EVM de cada subportadora. En la Figura 5.3 se muestran dichos valores para diferentes configuraciones del experimento. En dicha figura se puede observar cómo el EVM se degrada conforme aumenta la frecuencia (ó el índice de subportadora) de manera similar, pero no igual, a la respuesta en frecuencia del canal. Algo que llama la atención de esta gráfica es el EVM de la subportadoras centrales (índice 256 ó 1.25 GHz), el cual se degrada drásticamente haciendo que no se puedan transmitir muchos bits de información en dichas frecuencias. Esto se debe exclusivamente al ruido de reloj de muestreo del conversor AD que es inherente a los conversores entrelazados, tal como se ha sido explicado en el apartado 2.5. En la Figura 5.3 se incluyen los umbrales de EVM para los diferentes órdenes de modulación QAM considerando el uso de HD-FEC, estos umbrales se han de utilizar para la toma de decisión de la carga binaria de cada subportadora, tal como se ha descrito en el apartado 3.1.3.2.

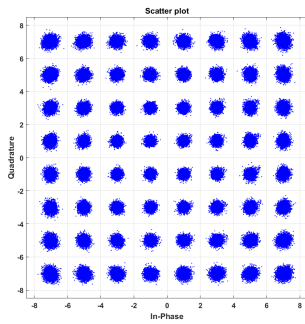
5.3 Medidas para carga constante

Una vez clara la respuesta en frecuencia y el EVM promedio obtenido para el banco de pruebas de la Figura 5.1, se procede a realizar un primer conjunto de mediciones variando el orden de modulación, donde todas las subportadoras se cargan con la misma modulación QAM. En este experimento se han utilizado 512 subportadoras ($N=1024$) de las cuales, 475 subportadoras son para datos y 36 subportadoras han sido anuladas (7 de baja frecuencia, 5 en el entorno de 1,25 GHz y 24 de alta frecuencia), dando como resultado un ancho de banda de 2.349 MHz. Las subportadoras de datos han sido moduladas con las siguientes símbolos M -QAM: 512, 256, 128, 64. El preámbulo de la trama OFDM y los algoritmos utilizados son los descritos en el capítulo 3. Por último, decir que el experimento se ha realizado utilizando una fibra óptica de 10 km de longitud y que para evitar el SCFO se ha utilizado la misma fuente de reloj para ambos conversores. En la Tabla 5.1 se presentan los resultados obtenidos y en la Figura 5.4 se muestran los diagramas de constelación del receptor.

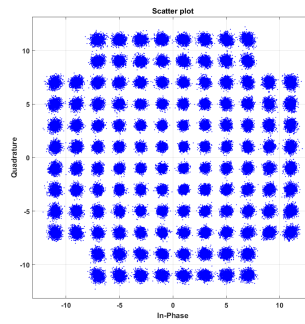
Para realizar la medida de velocidad se ha tomado como referencia un valor de BER de $2,4 \times 10^{-2}$ como umbral para obtener una transmisión libre de errores cuando se utiliza un SD-FEC y un 20 % de redundancia [101]; por otro lado, en caso de que se empleara un HD-FEC con un 7 % de redundancia, el umbral para la BER es de $3,8 \times 10^{-3}$ [83]. Los valores de BER obtenidos muestran que las señales moduladas con 64-, 128- y 256-QAM satisfacen el umbral para HD-FEC, sin embargo, cuando se utiliza la modulación 512-QAM

Tabla 5.1: Prestaciones del sistema propuesto utilizando diferentes modulaciones: 64-, 128-, 256- y 512-QAM.

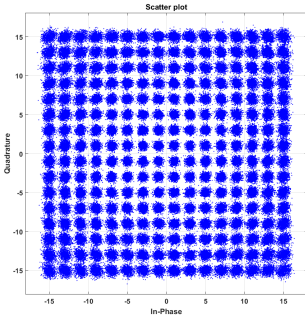
Modulación	Bitrate	BER	EVM max	EVM min	SE (bit/s/Hz)
64-QAM	13.70 Gb/s	3.69×10^{-8}	8 %	2 %	5.833
128-QAM	15.99 Gb/s	1.48×10^{-5}	9 %	2 %	6.808
256-QAM	18.27 Gb/s	2.26×10^{-3}	12 %	2 %	7.779
512-QAM	20.55 Gb/s	6.97×10^{-3}	12 %	2 %	8.750



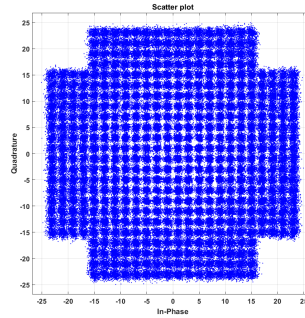
(a) 64-QAM



(b) 128-QAM



(c) 256-QAM



(d) 512-QAM

Figura 5.4: Diagramas de constelaciones para una transmisión sobre 10 km de fibra utilizando diferentes órdenes de modulación QAM.

se hace obligatorio el uso de un SD-FEC para realizar una transición exitosa sobre 10 km de SSMF. Las más altas tasas binarias y eficiencias espectrales se alcanzan con las modulaciones de 256- y 512-QAM, pero si se tiene en cuenta la redundancia introducida por cada uno de los correctores de errores, la tasa binaria libre de errores disminuye hasta los 16,44 Gb/s y 16,99 Gb/s para 512-QAM y 256-QAM, respectivamente. Esta pequeña diferencia en la tasa binaria y la significativa reducción en el coste *hardware* por utilizar un HD-FEC en lugar de un SD-FEC, hacen que modular las subportadoras de los símbolos OFDM con 256-QAM sea la mejor opción para este sistema de transmisión.

Ahora bien, si se toma el caso de la señal modulada con 256-QAM y transmitida sobre 10 km de SSMF de la Figura 5.3, se observa que a partir de la subportadora 420 en adelante utilizar esta modulación incumple con el umbral para HD-FEC. Sin embargo, en la Tabla 5.1 se muestra que la BER es aproximadamente la mitad de la de umbral. Esto que en un principio puede parecer confuso tiene una explicación muy simple: todas las subportadoras que están por debajo de la 420 contribuyen con muchos menos errores de los aceptables para este sistema y esto compensa el exceso de errores generado por el pequeño número de subportadoras utilizadas de más alta frecuencia. Si se profundiza un poco más en este análisis se puede decir que la cantidad de bits transmitidos por símbolo es $3.800 (8 \cdot 475)$, que la cantidad total de bits transmitido por trama es de $1.786.000 (3800 \cdot 470)$, que considerando una tasa de error de $3,8 \times 10^{-3}$ se pueden recibir como máximo 6.787 bits erróneos y que si esto se lo divide por el número de subportadoras que transmiten información se obtiene que en una trama cada subportadora puede contribuir con un máximo 13,25 bits erróneos.

En la parte superior de la Figura 5.5 se muestra la cantidad de bits erróneos recibidos por subportadora donde se observa que la gran mayoría de subportadoras reciben menos de 13,25 bits erróneos, y en la parte inferior se muestran los valores de BER por subportadora y el valor total de errores acumulados por un conjunto de subportadoras durante la recepción de los 475 símbolos OFDM. Como es de esperar, la mayor cantidad de errores, casi un 70 %, se concentran en las subportadoras cuyo índice es mayor a 420 (2 GHz aproximadamente). En la Figura 5.6 se representan los diagramas de constelación para 4 grupos de subportadoras y sus valores de EVM promedio para cada uno de los grupos. Como se puede ver, estos valores de EVM guardan relación con los de la Figura 5.3 para el caso de transmisión sobre 10 km de SSMF.

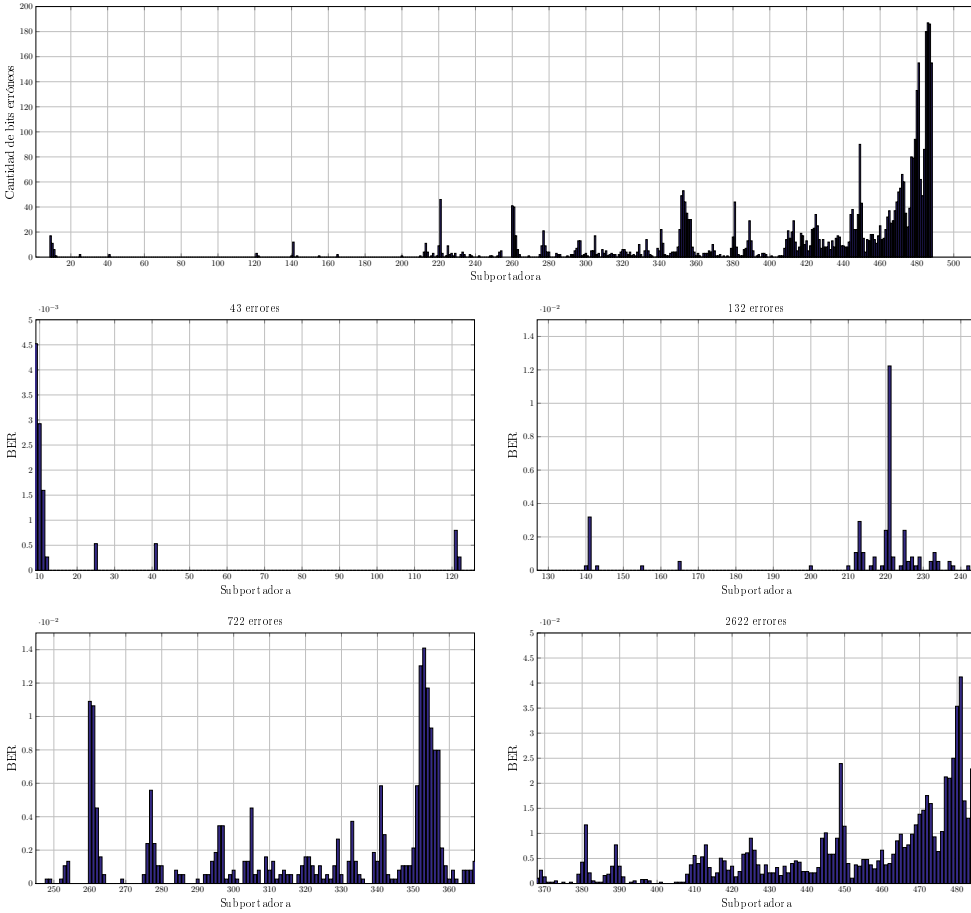


Figura 5.5: Cantidad de errores y BER por subportadora para una transmisión sobre 10 km de fibra utilizando una modulación 256-QAM.

5.4 Medidas para carga variable

De los experimentos anteriores quedan en claro dos cosas: que la modulación más eficiente para transmitir el mayor número de b/s es 256-QAM y que hay margen para utilizar otros órdenes de modulación, mayores y menores, en las algunas subportadoras. Haciendo énfasis en esto último, a continuación se presenta un experimento utilizando el algoritmo de *bit loading* descrito en el apartado 3.1.3.2, el cual busca llevar al máximo posible el orden de modulación asignado a cada subportadora sin que esto genere un exceso de bits erróneos en

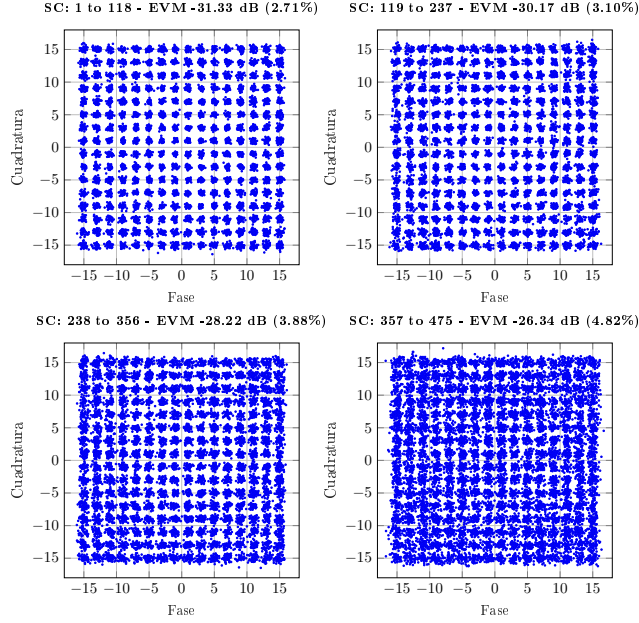


Figura 5.6: Diagramas de constelación para diferentes subconjuntos de subportadoras para una transmisión sobre 10 km de fibra utilizando una modulación 256-QAM.

el receptor. A partir de los valores de EVM obtenidos en el primer experimento, que se muestran en la Figura 5.3, y utilizando el algoritmo *bit loading* se obtiene la asignación de número de bits por subportadora que se representan en el gráfico 3D de la Figura 5.7.

Para todos los esquemas de transmisión se observa un patrón similar en las curvas de carga binaria: en la zona de baja frecuencia se presenta una subida escalonada muy abrupta, luego continua un gran ancho de banda donde se concentran los mayores órdenes de modulación, y finaliza con una zona de muy alta frecuencia donde se observa una bajada escalonada menos abrupta que la de baja frecuencia. Es de destacar que salvo para la transmisión sobre 40 km de fibra óptica, para todos los otros esquemas se alcanza a utilizar la modulación 512-QAM.

En este experimento se ha utilizado una FFT de 1024 puntos, lo que da un número de subportadoras (N_{sc}) de 512 debido a la simetría hermitiana. De las 512 subportadoras, 498 transportan datos (N_{usc}) y 13 se establecen nulas (4 en baja frecuencia y 9 en alta frecuencia). Las subportadoras de datos se

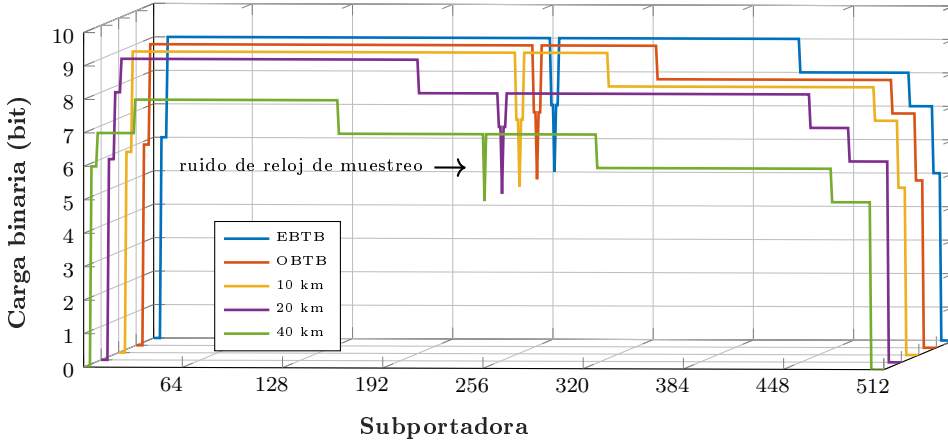


Figura 5.7: Carga binaria por subportadora para diferentes esquemas de transmisión.

modulan con símbolos M-QAM (16 a 512). Teniendo en cuenta la respuesta al impulso y el CD del enlace, la longitud del CP se establece en 16 muestras. Para reducir la PAPR de la señal OFDM, se aplica un recorte digital en la salida del transmisor OFDM. Se utiliza una fuente de reloj común para evitar el SCFO, pero la fase de ambos relojes no está alineada.

A continuación, en la Tabla 5.2, se resumen los resultados obtenidos usando las modulaciones QAM indicadas en la Figura 5.7 para las distintas configuraciones del sistema OFDM de tiempo real de la Figura 5.1. Se muestra la tasa binaria obtenida, la BER medida, la EVM medida tanto en la mejor como en la peor subportadora, y la SE alcanzada. El máximo valor de EVM para OBTB es 16 %, dicho valor es mayor que el obtenido en el caso de EBTB y esto se debe, en mayor medida, a que la incorporación de la etapa electro-óptica tiene una mayor atenuación a altas frecuencias. La Tabla 5.2 muestra que los valores de EVM mínimo crecen con la distancia de la fibra, ya que la SNR es peor para distancias más largas. Para el caso de 40 km, el valor de EVM mínimo es del 3,4 % y esto no permite el uso de una modulación 512-QAM para el caso de utilizar HD-FEC. Entonces, se emplea una orden de modulación de 256-QAM en las mejores subportadoras, dando como resultado una SE más baja.

Si se toma el caso de transmitir sobre 10 km de fibra óptica y se comparan los resultados tras modular todas las subportadoras con 256-QAM o según el algoritmo de carga binaria, ver Tabla 5.1 y Tabla 5.2, la tasa binaria se incrementa un 10 % y la SE aumenta casi un bit por el uso una carga eficiente de las subportadoras con un coste *hardware* despreciable.

Tabla 5.2: Prestaciones del sistema propuesto utilizando *bit-loading*

Config.	Bitrate	BER	EVM max	EVM min	SE (bit/s/Hz)
EBTB	20.86 Gb/s	3.43×10^{-3}	13%	2.2%	8.58
OBTB	20.47 Gb/s	3.64×10^{-3}	16%	2.2%	8.48
10 km	20.37 Gb/s	3.52×10^{-3}	17%	2.6%	8.38
20 km	19.63 Gb/s	3.62×10^{-3}	18%	2.6%	8.07
40 km	16.37 Gb/s	3.52×10^{-3}	19%	3.4%	6.73

5.5 Comparación con otros diseños

En la Tabla 5.3 se muestran las características y las prestaciones de 9 sistemas OFDM para comunicaciones ópticas con detección directa y sin ningún tipo de multiplexado (ni en frecuencia, ni en longitud de onda) encontrados en la literatura y se los comparan con el sistema que ha sido desarrollado a lo largo de esta tesis. Estos 10 sistemas se pueden clasificar en tres grupos dependiendo del uso o no de equipo de laboratorio para generar/capturar la señal eléctrica, o el uso o no de la técnica de sobremuestreo para reducir el ruido de cuantificación y lograr una ENOB más alto. El grupo A corresponde a los sistemas que usan FPGA+DAC/ADC sin técnica de sobremuestreo, el grupo B utiliza FPGA+DAC/ADC con técnica de sobremuestreo, y el grupo C emplea AWG y/o DSO, y técnica de sobremuestreo; los grupos A y B se corresponden con sistemas de tiempo real.

Tabla 5.3: Comparación de sistemas IM/DD OOFDM

Grupo	Ref.	DAC (GS/s)	ADC (GS/s)	Fs (GS/s)	M- QAM	N/N_{cp}	BW (MHz)	Bitrate (Gb/s) ^{7,8}	SE (bit/s/Hz) ^{7,8}	BER	FEC	Long. (km)	Mod. Opt.
A	Prop. ¹	5	5	5	16 to 512	1024/16	2432	20.37 / 18.95	8.38 / 7.79	3.8×10^{-3}	HD	10	DML
A	Prop. ¹	5	5	5	16 to 512	1024/16	2432	19.63 / 18.26	8.07 / 7.51	3.8×10^{-3}	HD	20	DML
A	Prop. ¹	5	5	5	16 to 512	1024/16	2432	16.86 / 15.68	6.73 / 6.26	3.8×10^{-3}	HD	40	DML
A	[57] ²	2	2	2	16 to 512	64/8	906	6.31 / 5.86	6.96 / 6.47	3.8×10^{-3}	HD	50	DML
C	[63] ¹⁴⁶	10	5	2.5	4096	2048/32	976	11.54 / 9.23	11.82 / 9.45	2.4×10^{-2}	SD	10	DML
C	[54] ¹⁵⁶	10	10	5	64	4096/4	2002	12.00 / 11.16	5.99 / 5.57	3.8×10^{-3}	HD	20.62	DML
B	[58] ¹²⁶	2.5	5	2.5	1024	128/16	625	5.56 / 5.17	8.89 / 8.27	3.8×10^{-3}	HD	20	EML
C	[56] ²⁵⁶	2.5	20	2.5	1024	128/16	625	5.56 / 4.44	8.89 / 7.11	2.4×10^{-2}	SD	20	EML
A	[4] ¹²	2.5	2.5	2.5	16 to 256	128/16	1094	6.32 / 5.88	5.78 / 5.37	1.9×10^{-3}	HD	20	EML
A	[60] ¹⁴	5	5	5	16 to 256	128/32	2150	10.44 / 9.71	4.85 / 4.51	3.8×10^{-3}	HD	20	EML
C	[53] ¹⁵⁶	4	20	4	256	256/14	1875	14.22 / 13.23	7.59 / 7.05	3.8×10^{-3}	HD	20	DML
A	[61] ³	4	4	4	64	32/8	2000	9.00 / 8.37	4.50 / 4.19	1.0×10^{-3}	HD	25	DML

¹ Virtex-7. ² Virtex-6. ³ Stratix II GX. ⁴ Generador arbitrario de funciones. ⁵ Osciloscopio digital. ⁶ Técnica de sobremuestreo. ⁷ Luego de excluir la redundancia del CP. ⁸ Luego de excluir la redundancia del CP y el FEC.

Según el conocimiento del estado del arte, el sistema presentado en [60] de 10,44 Gb/s con un SE de 4,85 b/s/Hz que transmite subportadoras moduladas desde 16 hasta 256-QAM a través de 20 km de fibra monomodo estándar usando HD-FEC es el de tasa binaria más alta que se ha publicado para un sistema IM/DD OOFDM, sin amplificador óptico, de longitud de onda única, de banda de frecuencia única, en tiempo real y de bajo coste. Hay otros sistemas con tasas de bits más altas donde el procesamiento no se realiza en tiempo real [54-56, 64], o usan componentes muy caros [65], o multiplexación por división de frecuencia [66, 67], o multiplexación por división de onda [68-70].

Solo hay 3 sistemas, [54], [55] y el propuesto en esta tesis, que alcanzan tasas binarias efectivas (es decir, después de excluir el CP y la redundancia introducida por el FEC) superiores a 10 Gb/s en 20 km SSMF con BW similar. En [54] los autores han utilizado una FFT de gran longitud y una modulación de bajo orden, por el contrario, en [55] los autores han optado por una FFT de pocos puntos y un orden de modulación alto, mientras que en el presente trabajo se ha utilizado una FFT de gran longitud y órdenes de modulación altos. Ambos trabajos [54, 55] utilizan sobremuestreo y equipos de laboratorio (AWG y DSO). El uso de sobremuestreo reduce el ruido de cuantificación, mejora la respuesta de frecuencia analógica de los conversores y relaja las restricciones para el diseño del filtro antialiasing. Aunque este enfoque es interesante como experimento de laboratorio para analizar el rendimiento del sistema, no es un enfoque válido para la implementación de un sistema en tiempo real de bajo coste donde el ancho de banda disponible de los conversores AD y DA generalmente se explotan al máximo. A pesar de todas estas ventajas, el sistema de tiempo real aquí propuesto logra una tasa binaria un 50 % más rápida aproximadamente y con un mayor eficiencia espectral que la de los sistemas [54, 55]. En el supuesto de que los sistemas [54, 55] se implementaran en dispositivos FPGA actuales, estos requerirían más *hardware* que el sistema objeto de esta tesis debido a dos razones: la primera es que utilizan la técnica DFT-spread [44] para reducir el PAPR de la señal OFDM transmitida y la segunda debido a la alta tasa de muestreo que obliga a paralelizar aún más los algoritmos para alcanzar frecuencias de trabajo admisibles dentro de las FPGAs.

Las referencias [58, 63] tienen una SE de 1 y 2 bit/s/Hz mayor que el sistema aquí propuesto, respectivamente, pero alcanzan una tasa binaria mucho menor. Además, en [63] se utiliza SD-FEC que es más complejo desde el punto de vista *hardware* que el HD-FEC. Su mayor SE proviene de emplear solo una parte del BW disponible del DAC (78 % y 50 %, respectivamente) y la técnica de sobremuestreo para mejorar la SNR efectiva, razones por lo cual se puede usar un orden de modulación más alto pero al mismo tiempo la F_s efectiva es menor,

lo cual da como resultado que la velocidad del sistema sea inferior al de aquí propuesto.

Al analizar los sistemas de tiempo real [4, 57, 60, 61] se observa que todos tienen una tasa de binaria más baja que el sistema aquí propuesto, en particular [4, 60, 61] tienen una SE más baja. La tasa binaria para un sistema OFDM, descrito en la ecuación 1.3, se puede incrementar aumentando todos o alguno de los siguientes 4 factores: la frecuencia de muestreo, el orden de modulación, el cociente N/N_{cp} y el cociente N_{usc}/N . Los sistemas RT anteriormente mencionados tienen valores más bajos para estos 4 factores que él aquí propuesto, a excepción de [60] que funciona con la misma frecuencia de muestreo. En esta tesis se ha buscado maximizar los 4 factores, teniendo en cuenta las limitaciones y los costos tecnológicos, para obtener la tasa de binaria más alta posible.

5.5.1 Sistemas de alto coste

Otros autores para obtener velocidades más altas en sistemas OOFDM de tiempo real utilizan componentes muy caros, o FDM, o WDM o una mezcla de algunos de ellos. El uso de dispositivos con gran ancho de banda analógico (> 10 GHz) y conversores con alta tasa de muestreo (> 30 GS/s) elevan los costes del sistema de manera muy significativa. Por otra parte, utilizar algún tipo de multiplexado (FDM o WDM) implica utilizar varios conversores AD y DA, varios láseres con diferentes longitudes de onda (WDM) o combinadores y mezcladores de RF (FDM), lo cual también encarece el sistema.

Algunos sistemas [68-70] utilizan WDM junto conversores de varias decenas de GS/s, que tienen un coste varias veces superior al aquí utilizado y requieren varias FPGAs para la transmisión, adquisición y procesado de las muestras, y además aplican sobremuestreo para mejorar las prestaciones. Es el caso del sistema [70] donde se informa que se ha logrado transmitir 56.4 Gb/s, por longitud de onda utilizada, a través de 20 km de SSMF utilizando una modulación 128-QAM y SD-FEC para una señal OFDM con un BW de 8 GHz en un sistema en tiempo real que modula la señal óptica mediante un modulador MZM. Luego de remover la redundancia introducida por el FEC, la velocidad del sistema disminuye hasta los 45,01 Gb/s. En el experimento descrito en [70] se utiliza un DAC de 64 GS/s con un ancho de banda analógico de 13 GHz y un ADC de 41 GS/s con un ancho de banda analógico de 16 GHz. Los anchos de banda del amplificador y el modulador óptico son de 25 y 28 GHz, respectivamente.

En [66, 67] se presentan sistemas que utilizan FDM con componentes de bajo coste que se ajustan al ancho de banda utilizado. En el caso de [66] se utilizan 3 DACs de 8 bits a 4 GS/s para generar 3 señales OFDM de banda base (BW de 2 GHz) que se suben en frecuencia utilizando un modulador IQ y un mezclador. Utilizando un modulador de electro absorción (EML), un fotodetector (PIN+TIA) de 12 GHz de BW y un esquema OBTB, se han obtenido 10,375 Gb/s para la primera banda y 8 Gb/s para las dos bandas superiores. En [67] se presenta un sistema que utiliza los mismos conversores, la misma F_s y los mismos modelos de FPGA que los del presente trabajo. Se generan dos bandas, cada una con un BW de 2,5 GHz, con la particularidad de que la segunda banda se obtiene utilizando el modo retorno a cero del DAC. Utilizando un DML y 22 km de SSMF los autores de [67] han obtenido una velocidad de 23,06 Gb/s⁸ y una SE de 4,61 bit/s/Hz⁸.

Si se comparan los resultados obtenidos en [66, 67] con los de esta tesis se observa que: [66] (BW=6 GHz, OBTB, 26,37 Gb/s y 4,39 bit/s/Hz) con un BW 2,5 veces mayor obtiene una mejora de velocidad del 30 % con una caída de la SE del 50 % y [67] (BW=5 GHz, 22 km, 23,06 Gb/s y 4,61 bit/s/Hz) con un BW 2 veces mayor obtiene una mejora de velocidad del 26 % con una caída de la SE del 37 %.

5.6 Análisis del desplazamiento de frecuencia de reloj de muestreo

En la práctica no es válido asumir que la diferencia entre la frecuencia del oscilador del transmisor y del receptor es nula. Aunque los osciladores tienen la misma frecuencia nominal, nunca son iguales. Esta diferencia genera un desplazamiento de la frecuencia del reloj de muestreo que necesita ser estimado y compensado por el receptor para evitar la degradación de las prestaciones. Los resultados anteriores se obtuvieron utilizando una fuente de reloj común para evitar el SCFO entre los dos conversores. En esta sección se ha realizado un análisis cuantitativo de cómo afectan distintos valores de desplazamiento de frecuencia a las prestaciones del sistema.

El efecto del SCFO ha sido analizado exhaustivamente en la literatura y se ha demostrado que puede causar principalmente tres tipos de problemas: la rotación de fase de la subportadora, la interferencia entre símbolos y la interferencia entre subportadoras. El ISI proviene del hecho de que los diferentes periodos de muestreo entre el transmisor y el receptor causan una deriva temporal de la muestra óptima desde la cual debe comenzar la ventana de FFT,

este efecto toma varios símbolos OFDM para volverse problemático y se puede estimar que mueve la ventana una muestra hacia adelante o atrás cuando la deriva de tiempo acumulada es significativa [133].

Por otro lado, la rotación de fase de subportadora y el ICI se ven en el dominio de frecuencia después de la FFT. El primero es un factor de rotación común en cada símbolo OFDM que crece linealmente con el índice de la subportadora k y con el índice de símbolos OFDM m [134]:

$$\phi_{km} = S \cdot k \cdot m \quad \text{con} \quad S = \frac{2\pi N_t \Delta}{N},$$

donde $N_t = N + N_{cp}$ y $\Delta = (f_t - f_r)/f_r$ es la diferencia de frecuencia normalizada entre osciladores (f_t y f_r son las frecuencias de muestreo del transmisor y receptor, respectivamente). Esta rotación de fase puede estimarse usando el preámbulo de entrenamiento y/o los pilotos incrustados en los símbolos OFDM [134]. Una vez estimada, se puede compensar rotando los coeficientes del ecualizador de canal en el dominio de la frecuencia por e^{-jSkm} . En el presente sistema, para el caso de $N=1024$, solo se necesita rotar una cuarta parte de los coeficientes del ecualizador, ya que el resto se puede obtener utilizando el interpolador lineal propuesto en el apartado 3.2.3. Este enfoque es similar al utilizado en [135]. La segunda distorsión en el dominio de frecuencia es la presencia de ICI debido a la pérdida de ortogonalidad entre las subportadoras [136], que depende de los datos transportados en cada subportadora y puede verse como una interferencia aleatoria. Además, crece con el índice de la subportadora, por lo que es mayor para los sistemas OFDM con N grandes.

Se han realizado algunas mediciones para verificar, cómo la diferencia entre los osciladores (10, 20 y 40 ppm), afecta a las prestaciones del sistema. Para cada medición se han enviado paquetes con 100 símbolos OFDM, la señal recibida ha sido procesada en MATLAB, donde el SCFO se ha estimado al comienzo de cada paquete y con dicha estimación se ha compensado la rotación de fase en el ecualizador de canal, tal como se describe en [135]. La Tabla 5.4 muestra el EVM medio por subportadora obtenido para los diferentes valores de SCFO, para mayor claridad, las subportadoras se han agrupado en cuatro grupos. Cada grupo cubre una cuarta parte de las 498 subportadoras activas y esto facilita poder comparar el rendimiento en diferentes rangos de frecuencias. Se puede ver que la compensación de rotación de fase funciona bien para 10 y 20 ppm, y los 4 grupos dan resultados similares. Por el contrario, para 40 ppm el EVM aumenta en mayor medida para el grupo de subportadoras de más alta frecuencia. En este caso el ICI es más alto y no puede ser compensado, causando

el incremento de EVM. Estos resultados muestran que el sistema presentado puede funcionar correctamente con SCFO hasta 40 ppm, solo que en este último caso se debe evaluar el orden de modulación de algunas subportadoras de alta frecuencia para verificar si deben reducirse debido a la distorsión de ICI.

Tabla 5.4: Mediciones de EVM para diferentes valores de SCFO

SCFO \ k	1-124	125-249	250-373	374-498
0 ppm	2.64%	2.91%	3.80%	4.98%
10 ppm	2.71%	3.01%	3.83%	5.21%
20 ppm	2.72%	2.99%	3.88%	5.22%
40 ppm	2.80%	3.14%	4.32%	6.13%

5.7 Conclusiones

En primer lugar, se ha validado el correcto funcionamiento del conjunto de algoritmos y sistemas de control desarrollados en esta tesis mediante la transmisión y recepción de miles de millones bits por segundo en tiempo real a través de un banco de pruebas óptico utilizando OFDM. Se ha obtenido la estimación del canal para las diferentes configuraciones experimentales (ver Figura 5.2) que se condicen con las expectativas generadas a partir de la caracterización de los diferentes elementos que componen este módem, obtenidas en el capítulo 2.

Mediante una carga homogénea de subportadoras se han realizado varios experimentos que han permitido obtener, entre otras cosas, el valor de EVM promedio para cada subportadora (ver Figura 5.3) y de manera indirecta su correspondiente valor de SNR [41]. Sobre un enlace óptico de 10 km de longitud se obtuvo que las mejores prestaciones se alcanzan al utilizar la modulación 256-QAM y que estos valores se pueden mejorar utilizando el algoritmo de carga binaria, un 10 % en velocidad y 1 bit en SE.

Existen varios trabajos en estos últimos años que han demostrado que es posible utilizar OFDM para la transmisión de datos a varios Gb/s sobre sistemas PON de detección directa. Algunos de ellos, utilizando equipos de laboratorio, trabajan con la señal OFDM sobremuestreada para mejorar la SNR de la porción del BW utilizado, lo cual se aleja de una solución de tiempo real de bajo coste. En la Tabla 5.3 figuran varios sistemas que utilizan sobremuestreo pero sin embargo ninguno supera la velocidad del sistema desarrollado en esta tesis, esto se debe a que la F_s efectiva y el BW utilizado es inferior. Algunos

de ellos alcanzan una mejor SE producto de emplear modulaciones más altas, las cuales pueden ser utilizadas gracias a la técnica de sobremuestreo. Si se comparan dos sistemas que utilizan la misma F_s efectiva y un BW similar, de los cuales uno utiliza sobremuestreo ([63]) y el otro no ([4]), se observa que las prestaciones del primero mejoran de manera significativa con relación al segundo.

En [70] a pesar de utilizar componentes con BW analógico extremadamente grande, sobremuestreo, una F_s equivalente de 16 GS/s y una señal con un BW de 8 GHz, la mejora de la tasa binaria es de 2,5 veces y la SE cae 1,33 veces en relación a los resultados obtenidos por el sistema aquí propuesto para el caso de 20 km utilizando HD-FEC. El uso de componentes con ancho de banda tan elevados, conversores de altísima frecuencia de muestreo, un modulador óptico externo y varios dispositivos FPGA, hacen que este tipo de soluciones se alejan enormemente de lo que se puede considerar sistema de bajo coste. Los sistemas presentados en [66, 67] obtienen una mejora de velocidad inferior al 30% y duplican o triplican el coste del presente trabajo.

En esta tesis, se ha demostrado que se puede lograr una alta velocidad de datos y una elevada SE utilizando componentes comerciales disponibles en el mercado de bajo coste gracias a la optimización de los 4 factores que rigen este tipo de sistemas.

Por último, se ha logrado un *bitrate* de 20,37 Gb/s utilizando carga adaptativa con modulaciones que van desde 32-QAM hasta 512-QAM y se ha transmitido con éxito sobre 10 km de SSMF con una SE de 8,38 bit/s/Hz. También se han alcanzado velocidades de 19,63/16,37 Gb/s para enlaces SSMF de 20/40 km con eficiencias de 8,07 y 6,73 bit/s/Hz. Se ha analizado la sensibilidad del sistema al SCFO entre DAC y ADC, y no se han observado diferencias significativas para desplazamientos de frecuencia de hasta 20 ppm.

Conclusiones y trabajos futuros

En este capítulo se exponen las principales conclusiones de la tesis y se plantean las líneas de investigación futuras.

6.1 Conclusiones

El objetivo de esta tesis ha sido el diseño, implementación y verificación experimental de un sistema OFDM para comunicaciones ópticas de tiempo real y de bajo coste utilizando dispositivos FPGA. La atenuación, el ancho de banda, la no linealidad y la distorsión de cada dispositivo que componen el sistema de comunicaciones ópticas desarrollado en esta tesis, afectan al EVM de cada subportadora y, por lo tanto, a las prestaciones del sistema. Debido a esto, la primera tarea de este trabajo ha sido el estudio en profundidad de cada uno de los elementos *hardware* que integran este módem, lo cual ha permitido conocer las limitaciones de los componentes y conocer su comportamiento a lo largo de todo el ancho de banda utilizado, los cuales condicionan el diseño del sistema. Este estudio arroja como resultado una respuesta en frecuencia principalmente del tipo paso bajos, con más de 10 dB de atenuación para las altas frecuencias, y con una débil respuesta paso altos. Se ha visto que una mala calibración analógica del ADC modifica las distorsiones del conversor de manera significativa, a tal punto que puedan llegar a afectar las prestaciones del sistema. Lo

aprendido tras este estudio de los elementos *hardware*, ha servido de punto de partida para las mejoras del sistema OOFDM desarrollado en esta tesis.

Para poder seleccionar y verificar los algoritmos de DSP, se ha desarrollado en MATLAB un modelo funcional (secuencial y en coma flotante) y un modelo *hardware* (codificación con funciones básicas y en punto fijo) del transmisor y receptor, así como también se ha modelado el comportamiento de los conversores y la etapa óptica. Este simulador ha permitido: acelerar la prueba de diferentes tipos de algoritmos, cuantificar la relación entre la precisión finita de los módulos de DSP y las prestaciones del sistema, y generar los vectores de datos para verificar la implementación sobre FPGA.

Para la sincronización temporal de la trama se ha propuesto un algoritmo nuevo que se aprovecha del uso de un preámbulo repetitivo, donde el receptor realiza una correlación cruzada entre la señal recibida y la parte repetitiva del preámbulo, usando sólo un bit para cuantificar ambas señales. El coste computacional del mismo es mucho menor al resto de los algoritmos estudiados. El diseño, implementación y verificación experimental de este TSA ha sido publicado en [130], donde se ha demostrado la viabilidad del algoritmo y sus buenas prestaciones en escenarios con baja SNR.

El bloque DSP con mayor coste computacional, y por lo tanto el que utiliza el mayor número de recursos de la FPGA, es sin duda el de la FFT. Se ha puesto especial cuidado en la selección del algoritmo, en su diseño e implementación y se han tomado las precauciones necesarias para disminuir la existencia de caminos críticos en este bloque. El hecho de ser un módulo FFT de alta velocidad y longitud configurable implica, por una parte, un incremento despreciable del coste *hardware* en lo que concierne estrictamente al algoritmo de FFT y, por otro lado, un desafío interesante para el diseño e implementación de los módulos de reordenamiento de datos necesarios para que los muestras entren y salgan del bloque en orden natural. Se ha demostrado que este módulo puede operar con un reloj de 625 MHz, lo cual abre las puertas a la posibilidad de trabajar con conversores de mayor tasa de muestreo (hasta 10 GS/s), con la natural consecuencia de aumentar la tasa binaria del sistema OOFDM. Para el reordenamiento de datos se ha propuesto una solución generalizable para diferentes bases del algoritmo, diferentes números de muestras en paralelo y diferentes longitudes de FFT; que además permite resolver tanto el reordenamiento de entrada como de salida.

En los sistemas de comunicación con fibra óptica, el canal puede ser considerado invariante en el tiempo, razón por la cual, un ecualizador *block-type* es una elección acertada. Para disminuir el tamaño del preámbulo, lo que implica

afectar lo menos posible a la tasa binaria, se ha utilizado un diezmado por 4 para el símbolo OFDM de estimación de canal. Para disminuir el ruido de la estimación se lo ha replicado 4 veces.

Se ha desarrollado un módem OFDM que trabaja en tiempo real a una tasa 5 GS/s utilizando dos tarjetas de evaluación que contienen un dispositivo FPGA Virtex-7, que se comunican a un ordenador mediante una interfaz gigabit ethernet y pueden ser controlados con total autonomía desde MATLAB de manera tal, que se pueden realizar pruebas de larga duración con diferentes características. Ha sido todo un desafío diseñar e implementar un sistema de tiempo real sobre FPGA, trabajando con una frecuencia de reloj principal de 312,5 MHz, múltiples dominios de reloj y 16 muestras en paralelo. Al mismo tiempo, esta implementación utiliza menos recursos *hardware* que los utilizados por el otro receptor OOFDM publicado con exactamente la misma tasa de muestreo.

Pensando en otras líneas de investigación futuras y aprovechando los módulos desarrollados para la puesta en marcha de los conversores AD y DA, se han implementado dos sistemas que permiten generar y capturar cualquier tipo de señal eléctrica que no necesite una tasa de muestreo superior a 5 GS/s y que puede ser procesada posteriormente en un ordenador. De esta manera será posible estudiar y analizar otro tipo de modulaciones (p. ej. monoportadora), nuevos algoritmos de DSP y nuevos enlaces ópticos (p. ej. luz visible).

Para poner a prueba el transmisor y receptor OFDM se ha desarrollado un banco de pruebas ópticos, el cual permite trabajar con fibra óptica de diferentes longitudes y adaptar los valores de la señal eléctrica a la entrada del modulador óptico y del ADC. Mediante una carga homogénea de subportadoras se han obtenido los valores promedio de EVM para cada subportadora (ver Figura 5.3) y esto ha servido como punto de partida para realizar la carga variable de cada subportadora (ver Figura 5.7).

Los sistemas IM/DD OFDM de bajo coste parecen ser el futuro para las nuevas generaciones de PONs y en este trabajo de tesis se ha demostrado que se puede lograr una alta tasa binaria y una elevada SE utilizando componentes comerciales de bajo coste. Hasta la fecha, el sistema IM/DD OOFDM PON de tiempo real, bajo coste, de una sola banda y una sola longitud de onda que presenta las mejores prestaciones es el publicado en [60]. Los autores han alcanzado un *bitrate* de 10,44 Gb/s y una SE de 4,85 bit/s/Hz, utilizando modulaciones de 16 a 256-QAM sobre 20 km de SSMF. En esta tesis doctoral se ha logrado un *bitrate* de 20,37 Gb/s y una SE de 8,38 bit/s/Hz utilizando carga adaptativa con modulación M -QAM, con valores de M desde 32 hasta 512, sobre 10 km

de SSMF. También se han alcanzado velocidades de 19,63/16,37 Gb/s para enlaces SSMF de 20/40 km con eficiencias de 8,07 y 6,73 bit/s/Hz. Todos estos resultados han sido publicados en [137].

6.2 Trabajos futuros

A continuación se proponen algunos posibles trabajos futuros para continuar la investigación desarrollada en esta tesis doctoral:

- **Multi-banda:** Incrementar la tasa binaria a través del uso de una frecuencia de muestreo más elevada (decenas de GS/s) implica incrementar mucho el coste de los conversores AD y DA, y de los dispositivos eléctricos y ópticos que integran el sistema. Una manera alternativa, eficiente y de bajo coste, es combinar múltiples bandas generadas con conversores de algunos GS/s operando en modo RZ (segunda y tercera ventana de Nyquist) sin la necesidad de utilizar mezcladores y osciladores de gran precisión, y para el receptor utilizar la técnica de submuestreo. El DAC MD657B, utilizado en este trabajo, cuenta con la posibilidad de trabajar en modo RZ. Hemos realizado algunas pruebas con fibra multi-núcleo y un modulador MZM que han sido publicadas en [138].
- **Moduladores ópticos:** Como se ha mencionado en esta tesis, existe principalmente tres moduladores de intensidad óptica: DML, EML y MZM. La idea es probar y comparar las prestaciones del sistema IM/DD PON OOFDM obtenidas con los tres moduladores utilizando técnicas de DSP para compensar las alinealidades y distorsiones de cada uno de ellos.
- **SSB-OFDM:** A medida que se aumenta la distancia y el ancho de banda, la dispersión cromática de la fibra comienza a afectar lo suficiente como para degradar las prestaciones del sistema IM/DD. Una solución a esto, es utilizar una modulación óptica de banda lateral única (SSB) que reduce el BW utilizado a la mitad, pero que presenta el inconveniente de la interferencia de batido entre señales (SSBI). Para disminuir la SSBI se pueden utilizar técnicas de DSP.
- **SCFO:** Como se ha comentado en este trabajo, siempre existe una diferencia de frecuencia de muestreo entre el transmisor y receptor debido al desvío de los osciladores de su frecuencia nominal. Existen varios métodos para estimar y compensar el SCFO en OFDM. La idea es probar algunas alternativas, ensayarlas e implementarlas dentro del sistema descrito en esta tesis.

Bibliografía

- [1] Huawei Technologies Co. *Technological Developments and Trends of Optical Networks*. White paper. 2016.
- [2] Cisco. *Cisco Visual Networking Index: Forecast and Methodology, 2016–2021*. White paper. 2017.
- [3] Y. Lee y B. Mukherjee. “Traffic engineering in next-generation optical Networks”. En: *IEEE Communications Surveys Tutorials* 6.3 (2004), págs. 16-33.
- [4] M. Chen, J. He, J. Tang, X. Wu y L. Chen. “Experimental demonstration of real-time adaptively modulated DDO-OFDM systems with a high spectral efficiency up to 5.76bit/s/Hz transmission over SMF links”. En: *Opt. Express* 22.15 (2014), págs. 17691-17699.
- [5] Y. m. Lin y P. l. Tien. “Next-generation OFDMA-based passive optical network architecture supporting radio-over-fiber”. En: *IEEE Journal on Selected Areas in Communications* 28.6 (2010), págs. 791-799.
- [6] N. Cvijetic. “OFDM for Next-Generation Optical Access Networks”. En: *Journal of Lightwave Technology* 30.4 (2012), págs. 384-398.
- [7] H. Song, B. W. Kim y B. Mukherjee. “Long-reach optical access networks: A survey of research challenges, demonstrations, and bandwidth

- assignment mechanisms”. En: *IEEE Communications Surveys Tutorials* 12.1 (2010), págs. 112-123.
- [8] P. D. Townsend, G. Talli, C. W. Chow, E. M. MacHale, C. Antony, R. Davey, T. D. Ridder, X. Z. Qiu, P. Ossieur, H. G. Krimmel y col. “Long Reach Passive Optical Networks”. En: *LEOS 2007 - IEEE Lasers and Electro-Optics Society Annual Meeting Conference Proceedings*. 2007, págs. 868-869.
- [9] R. P. Davey, D. B. Grossman, M. Rasztoivits-Wiech, D. B. Payne, D. Nettet, A. E. Kelly, A. Rafel, S. Appathurai y S. H. Yang. “Long-Reach Passive Optical Networks”. En: *Journal of Lightwave Technology* 27.3 (2009), págs. 273-291.
- [10] K. Y. Cho, U. H. Hong, Y. Takushima, A. Agata, T. Sano, M. Suzuki e Y. C. Chung. “103-Gb/s Long-Reach WDM PON Implemented by Using Directly Modulated RSOAs”. En: *IEEE Photonics Technology Letters* 24.3 (2012), págs. 209-211.
- [11] L. Shi, S. S. Lee, H. Song y B. Mukherjee. “Energy-Efficient Long-Reach Passive Optical Network: A Network Planning Approach Based on User Behaviors”. En: *IEEE Systems Journal* 4.4 (2010), págs. 449-457.
- [12] A. Larmo, M. Lindström, M. Meyer, G. Pelletier, J. Torsner y H. Wiemann. “The LTE link-layer design”. En: *IEEE Communications Magazine* 47.4 (2009), págs. 52-59.
- [13] European Telecommunications Standards Institute. *Digital video broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television*. ETSI EN 302 744 V1.5.1. 2004.
- [14] European Telecommunications Standards Institute. *Digital video broadcasting (DVB); transmission system for handheld terminals (DVB-H)*. ETSI EN 302 304 V1.1.1. 2004.
- [15] European Telecommunications Standards Institute. *Digital video broadcasting (DVB); framing structure, channel coding and modulation for satellite services to handheld devices (SH) below 3 GHz*. ETSI EN 302 583 V1.1.1. 2008.

-
- [16] European Telecommunications Standards Institute. *Digital video broadcasting (DVB); framing structure, channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)*. ETSI EN 302 755 V1.1.1. 2009.
- [17] Institute of Electrical and Electronics Engineers. *IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems*. IEEE 802.16-2009. 2009.
- [18] Institute of Electrical and Electronics Engineers. *IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: High Speed Physical Layer in the 5 GHz band*. IEEE 802.11a. 1999.
- [19] Institute of Electrical and Electronics Engineers. *IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE 802.11g. 2003.
- [20] International Telecommunication Union. *Physical layer management for digital subscriber line transceivers*. ITU-T Recommendation G.997.1. 2016.
- [21] H. Bülow, F. Buchali y A. Klekamp. “Electronic Dispersion Compensation”. En: *Lightwave Technology, Journal of* 26.1 (2008), págs. 158-167.
- [22] A. Singer, N. Shanbhag y H.-M. Bae. “Electronic dispersion compensation”. En: *Signal Processing Magazine, IEEE* 25.6 (2008), págs. 110-130.
- [23] J. M. Tang y K. A. Shore. “30-Gb/s signal transmission over 40-km directly modulated DFB-laser-based single-mode-fiber links without optical amplification and dispersion compensation”. En: *Journal of Lightwave Technology* 24.6 (2006), págs. 2318-2327.
- [24] T. Duong, N. Genay, M. Ouzzif, J. L. Masson, B. Charbonnier, P. Chanclou y J. C. Simon. “Adaptive Loading Algorithm Implemented in AMOOFDM for NG-PON System Integrating Cost-Effective and Low-

- Bandwidth Optical Devices”. En: *IEEE Photonics Technology Letters* 21.12 (2009), págs. 790-792.
- [25] S. L. Jansen, I. Morita, T. C. W. Schenk, N. Takeda y H. Tanaka. “Coherent Optical 25.8-Gb/s OFDM Transmission Over 4160-km SSMF”. En: *Journal of Lightwave Technology* 26.1 (2008), págs. 6-15.
- [26] W. Shieh e I. Djordjevic. *OFDM for Optical Communications*. First. ELSEVIER Inc., 2010.
- [27] I. B. Djordjevic y B. Vasic. “Orthogonal frequency division multiplexing for high-speed optical transmission”. En: *Opt. Express* 14.9 (2006), págs. 3767-3775.
- [28] W. Shieh e Y. Tang. “Ultrahigh-Speed Signal Transmission Over Nonlinear and Dispersive Fiber Optic Channel: The Multicarrier Advantage”. En: *IEEE Photonics Journal* 2.3 (2010), págs. 276-283.
- [29] R. W. Chang. “Synthesis of band-limited orthogonal signals for multichannel data transmission”. En: *The Bell System Technical Journal* 45.10 (1966), págs. 1775-1796.
- [30] R. W. Chang. *Orthogonal frequency multiplex data transmission system*. U.S.Patent 3 488 445. Bell Telephone Labor Inc, 1970.
- [31] S. Weinstein y P. Ebert. “Data Transmission by Frequency-Division Multiplexing Using the Discrete Fourier Transform”. En: *IEEE Transactions on Communication Technology* 19.5 (1971), págs. 628-634.
- [32] A. Peled y A. Ruiz. “Frequency domain data transmission using reduced computational complexity algorithms”. En: *ICASSP '80. IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 5. 1980, págs. 964-967.
- [33] J. Armstrong. “OFDM for Optical Communications”. En: *Lightwave Technology, Journal of* 27.3 (2009), págs. 189-204.

-
- [34] N. LaSorte, W. J. Barnes y H. H. Refai. “The History of Orthogonal Frequency Division Multiplexing”. En: *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*. 2008, págs. 1-5.
- [35] F. Buchali, R. Dischler y X. Liu. “Optical OFDM: A promising high-speed optical transport technology”. En: *Bell Labs Technical Journal* 14.1 (2009), págs. 125-146.
- [36] J. A. C. Bingham. “Multicarrier modulation for data transmission: an idea whose time has come”. En: *IEEE Communications Magazine* 28.5 (1990), págs. 5-14.
- [37] Y. Bao, Z. Li, J. Li, X. Feng, B. ou Guan y G. Li. “Nonlinearity mitigation for high-speed optical OFDM transmitters using digital pre-distortion”. En: *Opt. Express* 21.6 (2013), págs. 7354-7361.
- [38] C. Wei, H. Cheng, H. Chen, Y. Chen, H. Chu, K. Chang, I. Lu y J. Chen. “Analysis of Nonlinear Distortion and SSII Cancellation in EAM-Based IMDD OFDM Transmission”. En: *Journal of Lightwave Technology* 33.14 (2015), págs. 3069-3082.
- [39] C. Wei, H. Cheng y W. Huang. “On Adiabatic Chirp and Compensation for Nonlinear Distortion in DML-Based OFDM Transmission”. En: *Journal of Lightwave Technology* 36.16 (2018), págs. 3502-3513.
- [40] B. G. Kim, S. H. Bae, H. Kim e Y. C. Chung. “RoF-Based Mobile Front-haul Networks Implemented by Using DML and EML for 5G Wireless Communication Systems”. En: *Journal of Lightwave Technology* 36.14 (2018), págs. 2874-2881.
- [41] R. Schmogrow, B. Nebendahl, M. Winter, A. Josten, D. Hillerkuss, S. Koenig, J. Meyer, M. Dreschmann, M. Huebner, C. Koos y col. “Error Vector Magnitude as a Performance Measure for Advanced Modulation Formats”. En: *IEEE Photonics Technology Letters* 24.1 (2012), págs. 61-63.
- [42] S. H. Han y J. H. Lee. “An overview of peak-to-average power ratio reduction techniques for multicarrier transmission”. En: *IEEE Wireless Communications* 12.2 (2005), págs. 56-65.

- [43] P. Miao, D. Jiang, L. Wu y P. Chen. “A hybrid PAPR reduction approach for the IM/DD optical OFDM communications”. En: *2017 IEEE/CIC International Conference on Communications in China (ICCC)*. 2017, págs. 1-6.
- [44] Y. Tang, W. Shieh y B. S. Krongold. “DFT-Spread OFDM for Fiber Nonlinearity Mitigation”. En: *IEEE Photonics Technology Letters* 22.16 (2010), págs. 1250-1252.
- [45] Q. Yang, S. Chen, Y. Ma y W. Shieh. “Real-time reception of multi-gigabit coherent optical OFDM signals”. En: *Optics Express* 17.10 (2009), págs. 7985-7992.
- [46] S. D. Dissanayake y J. Armstrong. “Comparison of ACO-OFDM, DCO-OFDM and ADO-OFDM in IM/DD Systems”. En: *Journal of Lightwave Technology* 31.7 (2013), págs. 1063-1072.
- [47] B. Song, C. Zhu, B. Corcoran, Q. Wang, L. Zhuang y A. J. Lowery. “Experimental Layered/Enhanced ACO-OFDM Short-Haul Optical Fiber Link”. En: *IEEE Photonics Technology Letters* 28.24 (2016), págs. 2815-2818.
- [48] M. F. Sanya, L. Djogbe, A. Vianou y C. Aupetit-Berthelemot. “DC-biased optical OFDM for IM/DD passive optical network systems”. En: *IEEE/OSA Journal of Optical Communications and Networking* 7.4 (2015), págs. 205-214.
- [49] D. Patel, V. K. Singh y U. D. Dalal. “Assessment of the DC Bias to Mitigate the Clipping Noise in DCO-OFDM, ACO-OFDM; and Non-linear Distortion of DFB Laser Transmitted through Dispersive Single Mode Fibers in IM/DD Systems”. En: *Wireless Personal Communications* 96.1 (2017), págs. 341-360.
- [50] F. Barrami, Y. L. Guennec, E. Novakov y P. Busson. “An optical power efficient asymmetrically companded DCO-OFDM for IM/DD systems”. En: *2014 23rd Wireless and Optical Communication Conference (WOCC)*. 2014, págs. 1-6.
- [51] Q. Wang, B. Song, B. Corcoran, L. Zhuang y A. J. Lowery. “Real-Time Demonstration of Augmented-Spectral-Efficiency DMT Transmit-

- ter Using a Single IFFT”. En: *Journal of Lightwave Technology* 35.21 (2017), págs. 4796-4803.
- [52] M. Guo, J. Zhou, X. Tang, J. Qi, S. Liu, X. Xu, Y. Lu e Y. Qiao. “Optimized Layered/Enhanced ACO-SCFDM for Short-Reach Optical Interconnects”. En: *IEEE Photonics Technology Letters* 30.14 (2018), págs. 1321-1324.
- [53] E. Giacomidis, A. Kavatzikidis, A. Tsokanos, J. M. Tang e I. Tomkos. “Adaptive loading algorithms for IMDD optical OFDM PON systems using directly modulated lasers”. En: *IEEE/OSA Journal of Optical Communications and Networking* 4.10 (2012), págs. 769-778.
- [54] M. Chen, X. Xiao, Z. R. Huang, J. Yu, F. Li, Q. Chen y L. Chen. “Experimental Demonstration of an IFFT/FFT Size Efficient DFT-Spread OFDM for Short Reach Optical Transmission Systems”. En: *Journal of Lightwave Technology* 34.9 (2016), págs. 2100-2105.
- [55] F. Li, J. Yu, Y. Fang, Z. Dong, X. Li y L. Chen. “Demonstration of DFT-spread 256QAM-OFDM signal transmission with cost-effective directly modulated laser”. En: *Opt. Express* 22.7 (2014), págs. 8742-8748.
- [56] M. Chen, J. He y L. Chen. “Real-Time Demonstration of 1024-QAM OFDM Transmitter in Short-Reach IMDD Systems”. En: *IEEE Photonics Technology Letters* 27.8 (2015), págs. 824-827.
- [57] Q. Zhang, M. Liu, J. Chen, H. Dun, J. Zhang, Y. Li, Y. Song y M. Wang. “Real-time optical OFDM transmissions with spectral efficiency up to 6.93bit/s/Hz over 50km SSMF IMDD systems”. En: *Optics Communications* 387 (2017), págs. 12 -17.
- [58] M. Chen, J. He, Q. Fan, Z. Dong y L. Chen. “Experimental Demonstration of Real-Time High-Level QAM-Encoded Direct-Detection Optical OFDM Systems”. En: *Journal of Lightwave Technology* 33.22 (2015), págs. 4632-4639.
- [59] M. Chen, J. He y L. Chen. “Real-time optical OFDM long-reach PON system over 100 km SSMF using a directly modulated DFB laser”. En: *Optical Communications and Networking, IEEE/OSA Journal of* 6.1 (2014), págs. 18-25.

- [60] M. Chen, J. He, J. Tang, L. Chen y X. Wu. “Real-Time 10.4-Gb/s Single-Band Optical 256/64/16QAM Receiver for OFDM-PON”. En: *Photonics Technology Letters, IEEE* 26.20 (2014), págs. 2012-2015.
- [61] R. P. Giddings, X. Q. Jin, E. Hugues-Salas, E. Giacomidis, J. L. Wei y J. M. Tang. “Experimental demonstration of a record high 11.25Gb/s real-time optical OFDM transceiver supporting 25km SMF end-to-end transmission in simple IMDD systems”. En: *Opt. Express* 18.6 (2010), págs. 5541-5555.
- [62] X. Jin, R. Giddings, E. Hugues-Salas y J. Tang. “Real-time demonstration of 128-QAM-encoded optical OFDM transmission with a 5.25bit/s/Hz spectral efficiency in simple IMDD systems utilizing directly modulated DFB lasers”. En: *Opt. Express* 17.22 (2009), págs. 20484-20493.
- [63] Q. Chen, J. He, R. Deng, M. Chen y L. Chen. “FFT-Size Efficient 4096-QAM OFDM for Low-Cost DML-Based IMDD System”. En: *IEEE Photonics Journal* 8.5 (2016), págs. 1-10.
- [64] F. Li, X. Li, L. Chen, Y. Xia, C. Ge e Y. Chen. “High-Level QAM OFDM System Using DML for Low-Cost Short Reach Optical Communications”. En: *IEEE Photonics Technology Letters* 26.9 (2014), págs. 941-944.
- [65] F. Li, X. Xiao, X. Li y Z. Dong. “Real-time demonstration of DMT-based DDO-OFDM transmission and reception at 50 Gb/s”. En: *39th European Conference and Exhibition on Optical Communication (ECOC 2013)*. 2013, págs. 1-3.
- [66] R. P. Giddings, E. Hugues-Salas y J. M. Tang. “30Gb/s real-time triple sub-band OFDM transceivers for future PONs beyond 10Gb/s/λ;” en: *39th European Conference and Exhibition on Optical Communication (ECOC 2013)*. 2013, págs. 1-3.
- [67] R. Deng, J. He, J. Yu, Y. Wei, X. Xiao, K. Lv, X. Xin y G.-K. Chang. “Increasing data rate of an optical IMDD system using a cost-efficient dual-band transmission scheme based on RTZ DAC and sub-Nyquist sampling ADC”. En: *Opt. Express* 26.9 (2018), págs. 11599-11607.
- [68] F. Li, X. Xiao y J. Yu. “Realization of real-time 100G 16QAM OFDM signal detection”. En: *Proc.SPIE*. Vol. 9388. 2015, págs. 9388 -9388 -11.

-
- [69] F. Li, X. Xiao, J. Yu, J. Zhang y X. Li. “Real-time direct-detection of quad-carrier 200Gbps 16QAM-DMT with directly modulated laser”. En: *2015 European Conference on Optical Communication (ECOC)*. 2015, págs. 1-3.
- [70] F. Li, X. Xiao, J. Yu, X. Li, S. Shi, C. Ge, Y. Xia e Y. Chen. “Real-time reception of four channels 50 Gb/s class high-level QAM-DMT signal in short reach”. En: *2016 Optical Fiber Communications Conference and Exhibition (OFC)*. 2016, págs. 1-3.
- [71] C. Vogel y H. Johansson. “Time-interleaved analog-to-digital converters: status and future directions”. En: *2006 IEEE International Symposium on Circuits and Systems*. 2006, 4 pp.-3389.
- [72] B. Murmann. *ADC Performance Survey 1997-2018*. URL: <http://web.stanford.edu/~murmman/adcsurvey.html>.
- [73] C. Vogel. “The impact of combined channel mismatch effects in time-interleaved ADCs”. En: *IEEE Transactions on Instrumentation and Measurement* 54.1 (2005), págs. 415-427.
- [74] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara y K. Kobayashi. “Explicit analysis of channel mismatch effects in time-interleaved ADC systems”. En: *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 48.3 (2001), págs. 261-271.
- [75] V. T. D. Huynh, N. Noels, P. Rombouts, J. Armstrong y H. Steendam. “Effect of Time-Interleaved Analog-to-Digital Converter Mismatches on OFDM Performance”. En: *OFDM 2014; 18th International OFDM Workshop 2014 (InOWo'14); Proceedings of*. 2014, págs. 1-8.
- [76] S. Ponnuru, M. Seo, U. Madhow y M. Rodwell. “Joint Mismatch and Channel Compensation for High-Speed OFDM Receivers with Time-Interleaved ADCs”. En: *IEEE Transactions on Communications* 58.8 (2010), págs. 2391-2401.
- [77] R. A. Belcher. “ADC Standard IEC 60748-4-3: Precision Measurement of Alternative ENOB Without a Sine Wave”. En: *IEEE Transactions on Instrumentation and Measurement* 64.12 (2015), págs. 3183-3200.

- [78] T. E2V. *Application Note on Offset, Gain, Phase Calibration Sequence ADC Quad 10-bit 1.25G*. URL: <https://www.e2v.com/content/uploads/2014/09/Calibration.pdf>.
- [79] D. J. G. Mestdagh, P. M. P. Spruyt y B. Biran. "Effect of amplitude clipping in DMT-ADSL transceivers". En: *Electronics Letters* 29.15 (1993), págs. 1354-1355.
- [80] C. R. Berger, Y. Benlachtar y R. Killey. "Optimum Clipping for Optical OFDM with Limited Resolution DAC/ADC". En: *Advanced Photonics*. Optical Society of America, 2011, SPMB5.
- [81] X. Q. Jin, J. L. Wei, R. P. Giddings, T. Quinlan, S. Walker y J. M. Tang. "Experimental Demonstrations and Extensive Comparisons of End-to-End Real-Time Optical OFDM Transceivers With Adaptive Bit and/or Power Loading". En: *IEEE Photonics Journal* 3.3 (2011), págs. 500-511.
- [82] P. K. Vitthaladevuni, M. S. Alouini y J. C. Kieffer. "Exact BER computation for cross QAM constellations". En: *IEEE Transactions on Wireless Communications* 4.6 (2005), págs. 3039-3050.
- [83] International Telecommunication Union. *Forward error correction for high bit-rate DWDM submarine systems*. ITU-T Recommendation G.975.1. 2004.
- [84] H. Y. Chen, C. C. Wei, D. Z. Hsu, M. C. Yuang, J. Chen, Y. M. Lin, P. L. Tien, S. S. W. Lee, S. H. Lin, W. Y. Li y col. "A 40-Gb/s OFDM PON System Based on 10-GHz EAM and 10-GHz Direct-Detection PIN". En: *IEEE Photonics Technology Letters* 24.1 (2012), págs. 85-87.
- [85] R. B. Nunes, A. Shahpari, J. A. L. Silva, M. Lima, P. S. B. de André y M. E. V. Segatto. "Experimental Demonstration of a 33.5-Gb/s OFDM-Based PON With Subcarrier Pre-Emphasis". En: *IEEE Photonics Technology Letters* 28.8 (2016), págs. 860-863.
- [86] S. Y. Kim, J. Terada y N. Yoshimoto. "SNR optimization through digital frequency-domain pre-emphasis for extending transmission distances in DML-based OFDM-PON systems". En: *2013 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC)*. 2013, págs. 1-3.

-
- [87] D.-Z. Hsu, C.-C. Wei, H.-Y. Chen, J. Chen, M. C. Yuang, S.-H. Lin y W.-Y. Li. “21 Gb/s after 100 km OFDM long-reach PON transmission using a cost-effective electro-absorption modulator”. En: *Opt. Express* 18.26 (2010), págs. 27758-27763.
- [88] M. Clara. *High-Performance D/A-Converters*. First. Springer, 2013.
- [89] L. Yang y J. Armstrong. “Oversampling to reduce the effect of timing jitter on high speed OFDM systems”. En: *IEEE Communications Letters* 14.3 (2010), págs. 196-198.
- [90] Maxim Integrated. *Equalizing Techniques Flatten DAC Frequency Response*. Application Note 3853. 2012.
- [91] T. Schmidl y D. Cox. “Robust frequency and timing synchronization for OFDM”. En: *Communications, IEEE Transactions on* 45.12 (1997), págs. 1613-1621.
- [92] B. Park, H. Cheon, C. Kang y D. Hong. “A novel timing estimation method for OFDM systems”. En: *Communications Letters, IEEE* 7.5 (2003), págs. 239-241.
- [93] M. Canet, V. Almenar, S. Flores y J. Valls. “Low Complexity Time Synchronization Algorithm for OFDM Systems with Repetitive Preambles”. En: *Journal of Signal Processing Systems* 68.3 (2012), págs. 287-301.
- [94] X. Q. Jin, R. P. Giddings, E. Hugues-Salas, and J. M. Tang. “Real-time experimental demonstration of optical OFDM symbol synchronization in directly modulated DFB laser-based 25km SMF IMDD systems”. En: *Opt. Express* 18.20 (2010), págs. 21100-21110.
- [95] X. Jin y J. Tang. “Optical OFDM Synchronization With Symbol Timing Offset and Sampling Clock Offset Compensation in Real-Time IMDD Systems”. En: *Photonics Journal, IEEE* 3.2 (2011), págs. 187-196.
- [96] N. Kaneda, Q. Yang, X. Liu, S. Chandrasekhar, W. Shieh e Y.-k. Chen. “Real-Time 2.5 GS/s Coherent Optical Receiver for 53.3-Gb/s Sub-Banded OFDM”. En: *Lightwave Technology, Journal of* 28.4 (2010), págs. 494-501.

- [97] S. Chen, Q. Yang y W. Shieh. “Demonstration of 12.1-Gb/s single-band real-time coherent optical OFDM reception”. En: *OptoElectronics and Communications Conference (OECC), 2010 15th.* 2010, págs. 472-473.
- [98] M. Chen, J. He, Z. Cao, J. Tang, L. Chen y X. Wu. “Symbol synchronization and sampling frequency synchronization techniques in real-time DDO-OFDM systems”. En: *Optics Communications* 326.0 (2014), págs. 80 -87.
- [99] A. Troya, K. Maharatna, M. Krstic, E. Grass, U. Jagdhold y R. Kraemer. “Low-Power VLSI Implementation of the Inner Receiver for OFDM-Based WLAN Systems”. En: *Circuits and Systems I: Regular Papers, IEEE Transactions on* 55.2 (2008), págs. 672-686.
- [100] K. K. Parhi. *VLSI Digital Signal Processing Systems: Design and Implementation*. First. Wiley-Interscience, 1999, pp. i-808.
- [101] L. Nelson, G. Zhang, M. Birk, C. Skolnick, R. Isaac, Y. Pan, C. Rasmussen, G. Pendock y B. Mikkelsen. “A robust real-time 100G transceiver With soft-decision forward error correction [Invited]”. En: *Optical Communications and Networking, IEEE/OSA Journal of* 4.11 (2012), B131-B141.
- [102] J. Cooley y J. Tukey. “An algorithm for the machine calculation of complex Fourier series”. En: *Mathematics of Computation* 19.90 (1965), págs. 297-301.
- [103] G. Bergland. “Fast Fourier transform hardware implementations - An overview”. En: *Audio and Electroacoustics, IEEE Transactions on* 17.2 (1969), págs. 104-108.
- [104] M. Jamali, J. Downey, N. Wilikins, C. Rehm y J. Tipping. “Development of a FPGA-based high speed FFT processor for wideband Direction of Arrival applications”. En: *Radar Conference, 2009 IEEE.* 2009, págs. 1-4.
- [105] G. Polat, S. Ozturk y M. Yakut. “Design and Implementation of 256-Point Radix-4 100 Gbit/s FFT Algorithm into FPGA for High-Speed Applications”. En: *ETRI Journal* 37.4 (2015), págs. 667-676.

-
- [106] J. Kim, J. Lee y K. Cho. “Design of 256-point FFT processor for 100 Gb/s coherent optical OFDM system”. En: *2016 IEEE International Symposium on Consumer Electronics (ISCE)*. 2016, págs. 61-62.
- [107] M. Garrido, M. Sánchez, M. L. López-Vallejo y J. Grajal. “A 4096-Point Radix-4 Memory-Based FFT Using DSP Slices”. En: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25.1 (2017), págs. 375-379.
- [108] Z. G. Ma, X. B. Yin y F. Yu. “A Novel Memory-Based FFT Architecture for Real-Valued Signals Based on a Radix-2 Decimation-In-Frequency Algorithm”. En: *IEEE Transactions on Circuits and Systems II: Express Briefs* 62.9 (2015), págs. 876-880.
- [109] H. L. Groginsky y G. Works. “A Pipeline Fast Fourier Transform”. En: *Computers, IEEE Transactions on* C-19.11 (1970), págs. 1015-1019.
- [110] J. A. Johnston. “Parallel pipeline fast Fourier transformer”. En: *Communications, Radar and Signal Processing, IEE Proceedings F* 130.6 (1983), págs. 564-572.
- [111] K.-J. Yang, S.-H. Tsai y G. Chuang. “MDC FFT/IFFT Processor With Variable Length for MIMO-OFDM Systems”. En: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 21.4 (2013), págs. 720-731.
- [112] M. Garrido, J. Grajal, M. Sanchez y O. Gustafsson. “Pipelined Radix- 2^k Feedforward FFT Architectures”. En: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 21.1 (2013), págs. 23-32.
- [113] M. Garrido, M. Acevedo, A. Ehliar y O. Gustafsson. “Challenging the limits of FFT performance on FPGAs (Invited paper)”. En: *2014 International Symposium on Integrated Circuits (ISIC)*. 2014, págs. 172-175.
- [114] H. S. Kang, S. H. Chang, I. K. Hwang y J. K. Lee. “A design and implementation of 32-paths parallel 256-point FFT/IFFT for optical OFDM systems”. En: *2016 18th International Conference on Advanced Communication Technology (ICACT)*. 2016, págs. 1-1.

- [115] G. R. Locharla, K. K. Mahapatra y S. Ari. “Variable length mixed radix MDC FFT/IFFT processor for MIMO-OFDM application”. En: *IET Computers Digital Techniques* 12.1 (2018), págs. 9-19.
- [116] S. Chen, S. Huang, M. Garrido y S. Jou. “Continuous-flow Parallel Bit-Reversal Circuit for MDF and MDC FFT Architectures”. En: *IEEE Transactions on Circuits and Systems I: Regular Papers* 61.10 (2014), págs. 2869-2877.
- [117] J. G. Proakis y D. G. Manolakis. *Digital signal processing: principles, algorithms, and applications*. Fourth. Prentice Hall, 2006.
- [118] B. Gold y T. Bially. “Parallelism in fast Fourier transform hardware”. En: *Audio and Electroacoustics, IEEE Transactions on* 21.1 (1973), págs. 5-16.
- [119] P. D. Welch. “A fixed-point fast Fourier transform error analysis”. En: *Audio and Electroacoustics, IEEE Transactions on* 17.2 (1969), págs. 151-157.
- [120] A. V. Oppenheim y C. J. Weinstein. “Effects of finite register length in digital filtering and the fast Fourier transform”. En: *Proceedings of the IEEE* 60.8 (1972), págs. 957-976.
- [121] Tran-thong y B. Liu. “Fixed-point fast Fourier transform error analysis”. En: *Acoustics, Speech and Signal Processing, IEEE Transactions on* 24.6 (1976), págs. 563-573.
- [122] W. Knight y R. Kaiser. “A simple fixed-point error bound for the fast Fourier transform”. En: *Acoustics, Speech and Signal Processing, IEEE Transactions on* 27.6 (1979), págs. 615-620.
- [123] S. Qadeer y M. Khan. “Fixed Point error analysis of Radix-4 and Radix-8 FFT algorithms”. En: *Multimedia, Signal Processing and Communication Technologies (IMPACT), 2011 International Conference on*. 2011, págs. 32-35.
- [124] O. Sarbishei y K. Radecka. “Analysis of Mean-Square-Error (MSE) for fixed-point FFT units”. En: *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*. 2011, págs. 1732-1735.

-
- [125] M. Mohammadnia y L. Shannon. “Minimizing the error: A study of the implementation of an Integer Split-Radix FFT on an FPGA for medical imaging”. En: *Field-Programmable Technology (FPT), 2012 International Conference on*. 2012, págs. 360-367.
- [126] S. Coleri, M. Ergen, A. Puri y A. Bahai. “Channel estimation techniques based on pilot arrangement in OFDM systems”. En: *IEEE Transactions on Broadcasting* 48.3 (2002), págs. 223-229.
- [127] L. Liu, X. Yang y W. Hu. “Experimental evaluation of pilot pattern design in direct-detection optical OFDM transmission”. En: *Optics Communications* 294 (2013), págs. 83 -87.
- [128] J. Heiskala y J. Terry. *OFDM Wireless LANs: A Theoretical and Practical Guide*. First. Sams Publishing, 2001.
- [129] H. G. Yeh y H. Seo. “Low complexity demodulator for M-ary QAM”. En: *2007 Wireless Telecommunications Symposium*. 2007, págs. 1-6.
- [130] J. S. Bruno, V. Almenar, J. Valls y J. L. Corral. “Low-Complexity Time Synchronization Algorithm for Optical OFDM PON System Using a Directly Modulated DFB Laser”. En: *J. Opt. Commun. Netw.* 7.11 (2015), págs. 1025-1033.
- [131] Xilinx, Inc. *16-Channel, DDR LVDS Interface with Per-Channel Alignment*. Application Note XAPP855. 2006.
- [132] M. Chen, R. Deng, Q. Chen, J. He y L. Chen. “Real-time system based on FPGA for optical communication system”. En: *Proc.SPIE* 10560 (2018), págs. 10560 -10560 -16.
- [133] M. Chen, J. He, J. Tang y L. Chen. “Pilot-aided sampling frequency offset estimation and compensation using DSP technique in DD-OFDM systems”. En: *Optical Fiber Technology* 20.3 (2014), págs. 268 -273.
- [134] Z. Zhang, Q. Zhang, Y. Li, Y. Song, J. Zhang y J. Chen. “A Single Pilot Subcarrier-Based Sampling Frequency Offset Estimation and Compensation Algorithm for Optical IMDD OFDM Systems”. En: *IEEE Photonics Journal* 8.5 (2016), págs. 1-9.

- [135] R. Deng, J. He, M. Chen, Y. Liu y L. Chen. “Real-Time LR-DDO-OFDM Transmission System Using EML With 1024-Point FFT”. En: *IEEE Photonics Technology Letters* 27.17 (2015), págs. 1841-1844.
- [136] T. Pollet, P. Spruyt y M. Moeneclaey. “The BER performance of OFDM systems using non-synchronized sampling”. En: *1994 IEEE GLOBECOM. Communications: The Global Bridge*. 1994, 253-257 vol.1.
- [137] J. S. Bruno, V. Almenar, J. Valls y J. L. Corral. “Real-time 20.37 Gb/s optical OFDM receiver for PON IM/DD systems”. En: *Opt. Express* 26.15 (2018), págs. 18817-18831.
- [138] M. Morant, J. S. Bruno, V. Almenar y R. Llorente. “Performance evaluation of OFDM and SC-QAM backhaul provision on FTTH optical access networks including multi-core fiber riser”. En: *Proc. SPIE 10945, Broadband Access Communication Technologies XIII* (2019), págs. 10945-8.