



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
ELÉCTRICA – TELECOMUNICACIONES

**Implementación de un Receptor OFDM IEEE 802.11
basado en SDR**

T E S I S
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:
CESAR MAYRHONT AVILA MORALES

TUTOR PRINCIPAL
VICTOR RANGEL LICEA, FACULTAD DE INGENIERÍA

MÉXICO, D.F., AGOSTO 2015

JURADO ASIGNADO:

Presidente: Dr. Gutiérrez Castrejón Ramón
Secretario: Dr. Matías Maruri José María
Vocal: Dr. Rangel Licea Víctor
1 er. Suplente: Dr. Moctezuma Flores Miguel
2 do. Suplente: Dr. Vicente Vivas Esaú

Lugar donde se realizó la tesis:

México D.F., Facultad de Ingeniería, UNAM.

TUTOR DE TESIS:

Dr. Víctor Rangel Licea

FIRMA

AGRADECIMIENTOS

A mis padres Francisco Ávila y Ma. Guadalupe Morales, por ser mi guía y mi mayor motivación, este logro es de ustedes, gracias por sus sabios consejos y por todo el apoyo que me han brindado a lo largo de mi vida, todo lo que soy es gracias a ustedes, los amo.

A mi hermano Paco, mi cuñada Julisa y mi sobrino Kalid, por siempre brindarme palabras de aliento y motivarme a salir adelante, gracias por acompañarme en cada paso importante de mi vida.

A mi asesor Dr. Victor Rangel Licea por sus enseñanzas, su dedicación y paciencia para el desarrollo de éste trabajo de tesis, gracias por brindarme la oportunidad de formar parte de su grupo de trabajo.

Al Dr. Robert Morelos Zaragoza por abrirme las puertas de su laboratorio en la universidad de San José, California. Por su hospitalidad y apoyo para mejorar el presente trabajo de tesis.

A la DGAPA-UNAM por el apoyo otorgado para la realización de ésta tesis mediante el proyecto PAPIIT No. IN114713 titulado “Diseño y análisis de algoritmos de calendarización en redes LTE y WiMAX”.

Finalmente mi infinito agradecimiento a la Universidad Nacional Autónoma de México por darme la oportunidad de continuar con mi desarrollo personal y profesional a través de sus aulas, hoy me siento orgulloso de haber concluido una de las etapas más importantes de mi vida en la máxima casa de estudios y prometo siempre poner en alto el nombre de esta institución.



RESUMEN

Hoy en día las comunicaciones inalámbricas juegan un papel fundamental en nuestra vida cotidiana, debido a que permiten la transmisión de datos, voz y video de una manera eficiente, sin necesidad del uso de cables, y cada vez a mayor velocidad.

Para el caso de las comunicaciones inalámbricas en lugares geográficos reducidos se creó el estándar IEEE 802.11 en el año de 1997, con el objetivo de definir las características de una red inalámbrica de área local (WLAN), específicamente sus funciones de capa PHY y de capa MAC. Desde su creación, el estándar IEEE 802.11 se ha ido mejorando y se ha convertido en una de las especificaciones más importantes en el campo de las telecomunicaciones.

Por otra parte, la tecnología de radios definidos por software (SDR) permite que las funciones más importantes de un radio tales como la sincronización, la modulación, codificación y otras etapas de capa física sean definidas vía software, por lo que los sistemas SDR son reconfigurables y permiten la implementación de distintas tecnologías en un mismo equipo.

Éste trabajo de tesis presenta el desarrollo de un sistema de recepción que une las dos tecnologías mencionadas anteriormente, es decir, un sistema SDR (USRP N210) que realiza las funciones de capa física definidas en el estándar IEEE 802.11 vía software (LabVIEW). Una vez implementado el sistema de recepción en la plataforma SDR, se evalúo su comportamiento como parte de un sistema de comunicaciones (transmisor y receptor) para medir el número de bits en error BER para distintos tipos de escenarios, dos en interiores y uno en exteriores, en función de la distancia entre transmisor - receptor y en función de la relación señal a ruido, además el análisis se llevó a cabo con distintos esquemas de modulación (BPSK y 64-QAM).



CONTENIDO

LISTA DE FIGURAS.....	VI
LISTA DE TABLAS.....	X
ACRÓNIMOS.....	XI
CAPITULO I INTRODUCCIÓN.....	1
1.1 Antecedentes.....	1
1.2 Definición del problema.....	2
1.3 Objetivos.....	4
1.4 Metodología.....	5
1.5 Contribuciones y Relevancia.....	6
1.6 Estructura de la tesis.....	6
CAPITULO II ESTADO DEL ARTE.....	8
2.1 Redes Wi-Fi.....	8
2.1.1 Introducción.....	8
2.1.2 Ventajas de las redes WLAN.....	9
2.1.3 La especificación IEEE 802.11.....	10
2.2 Trabajo Relacionado.....	12
CAPITULO III EL ESTÁNDAR IEEE 802.11.....	17
3.1 Introducción.....	17
3.1.1 La Familia de estándares IEEE 802.....	17
3.1.2 La topología de las redes IEEE 802.11.....	18
3.2 Capa MAC del estándar IEEE 802.11.....	20
3.2.1 Entrega de datos confiable.....	20



3.2.2 Control de Acceso al Medio.....	21
3.2.3 Formato de la trama MAC.....	24
3.3 Capa PHY del estándar IEEE 802.11.....	27
3.3.1 Arquitectura de la capa PHY.....	27
3.3.2 Esquemas de transmisión.....	27
3.4 El estándar IEEE 802.11g.....	29
3.4.1 Especificación ERP.....	29
3.4.2 OFDM.....	30
3.4.3 Descripción de la capa PHY para el esquema OFDM.....	34
 CAPÍTULO IV USRP N210.....	 44
4.1 Plataforma SDR.....	44
4.1.1 Introducción.....	44
4.1.2 Diagrama a bloques.....	46
4.1.3 Clasificación.....	47
4.2 USRP N210.....	49
4.2.1 Arquitectura.....	49
4.2.2 Tarjeta Madre.....	52
4.2.3 Tarjeta Hija SBX.....	53
 CAPÍTULO V SOFTWARE LabVIEW.....	 54
5.1 El entorno de programación de LabVIEW.....	54
5.1.1 Introducción.....	54
5.1.2 Instrumentos Virtuales.....	55
5.2 Programación del USRP N210 con LabVIEW.....	66
 CAPÍTULO VI DESARROLLO.....	 68
6.1 Arquitectura del receptor OFDM.....	68
6.2 Procesamiento de la señal en banda base.....	69
6.2.1 Diagrama a bloques.....	69
6.2.2 Etapa de sincronización.....	69



6.2.3 Etapa de ecualización.....	75
6.2.4 Decodificación del campo de señal.....	79
6.2.5 Decodificación del campo de datos.....	82
CAPÍTULO VII RESULTADOS.....	85
7.1 Comprobación de los bloques de procesamiento en banda base.....	85
7.1.1 Selector de Datos.....	85
7.1.2 Alineación del símbolo.....	87
7.1.3 Estimación del canal y ecualización.....	88
7.1.4 Corrección de fase.....	91
7.1.5 Demodulación.....	94
7.2 Pruebas del enlace inalámbrico con un Tx y Rx OFDM.....	94
7.2.1 Especificaciones del sistema.....	94
7.2.2 Escenario 1.....	97
7.2.3 Escenario 2.....	101
7.2.4 Escenario 3.....	105
CONCLUSIONES.....	108
TRABAJO FUTURO.....	111
REFERENCIAS.....	112
GLOSARIO.....	115
ANEXOS.....	118
ANEXO A Selector de Datos.....	118
ANEXO B Corrector de Fase.....	119
ANEXO C Deinterleaver.....	120
ANEXO D Demodulador.....	121
ANEXO E Depuncturing.....	123
ANEXO F Descrambler.....	124
ANEXO G Código para medición del BER.....	125



LISTA DE FIGURAS

3.1. Familia de Estándares IEEE 802 y su relación con el modelo OSI.....	18
3.2. Arquitectura del estándar IEEE 802.11.....	19
3.3. Distintas duraciones del intervalo IFS.....	23
3.4. Formato general de una trama MAC.....	24
3.5. Estructura del campo Frame Control.....	25
3.6. Estructura del estándar IEEE 802.11.....	27
3.7. Versiones del estándar IEEE 802.11 para redes WLAN.....	29
3.8. Comparación del uso de ancho de banda para sistemas FDM y OFDM... 3.9. Inserción del ciclo prefijo en sistemas OFDM.....	31 32
3.10. Diagrama a bloques de un transmisor OFDM.....	33
3.11. Diagrama a bloques de un receptor OFDM.....	34
3.12. Formato de la trama PPDU.....	34
3.13. Secuencias de entrenamiento que conforman el preámbulo PLCP	35
3.14. Estructura del campo SIGNAL con longitud de 24 bits.....	36
3.15. Estructura del campo de datos.....	37
3.16. Scrambler de datos.....	39
3.17. Diagrama del Codificador convolucional,r=1/2.....	40
3.18. Proceso de Puncturing para r=2/3.....	41
3.19. Constelaciones para modulación BPSK y 16-QAM.....	42
4.1. Diagrama a bloques de un sistema SDR.....	46
4.2. Imagen del equipo USRP N210.....	49
4.3. Diagrama a bloques del USRP N210.....	49
4.4. Panel Frontal del USRP N210.....	50
4.5. Tarjeta Madre del USRP N210.....	52



4.6. Tarjeta hija SBX.....	53
5.1. Panel Frontal de un VI en LabVIEW.....	56
5.2. Paleta de controles.....	56
5.3. Indicadores numéricos.....	57
5.4. Controles numéricos.....	57
5.5. Controles booleanos.....	57
5.6. Diagrama a bloques de un VI en LabVIEW.....	58
5.7. Paleta de funciones.....	61
5.8. Funciones numéricas.....	62
5.9. Funciones booleanas.....	62
5.10. Funciones de comparación.....	63
5.11. Funciones con arreglos.....	63
5.12. Funciones con cadenas de caracteres.....	64
5.13. Estructuras en LabVIEW.....	65
5.14. Panel de conexión (a) e Icono de un VI (b) en LabVIEW.....	65
5.15. Bloque niUSRP Open Rx Session.....	66
5.16. Bloque niUSRP Configure Signal.....	66
5.17. Bloque niUSRP Initiate.....	67
5.18. Bloque niUSRP Fetch Rx Data.....	67
5.19. Bloque niUSRP Abort.....	67
5.20. Bloque niUSRP Close Session.....	67
6.1. Diagrama a bloques del receptor OFDM.....	69
6.2. Bloques que conforman la etapa de sincronización.....	70
6.3. Implementación del bloque de corrección en frecuencia.....	72
6.4. Implementación del bloque de alineación del símbolo.....	74
6.5. Diagrama a bloques de la etapa de ecualización.....	75
6.6. Efecto de un canal selectivo en frecuencia.....	76
6.7. Implementación de estimación de canal y ecualización.....	77
6.8. Distribución de las portadoras en un símbolo OFDM	78
6.9. Diagrama a bloques de la etapa de decodificación.....	79
6.10. Demodulación del campo de señal.....	80



6.11. Implementación de la etapa de decodificación.....	82
6.12. Decodificación de la información.....	82
6.13. VI usado para la decodificación convolucional de los datos.....	84
7.1. Entrada de datos provenientes del USRP.....	86
7.2. Salida del bloque de Selección de Datos.....	86
7.3. Correlación entre muestras de entrada y secuencia de entrenamiento larga.....	87
7.4. Ejemplo de la estimación del canal de una trama OFDM recibida.....	88
7.5. Portadoras piloto después de la etapa de ecualización.....	89
7.6. Comparación de la constelación BPSK sin ecualización (izquierda) y con ecualización (derecha).....	90
7.7. Comparación de la constelación QPSK sin ecualización (izquierda) y con ecualización (derecha).....	90
7.8. Comparación de la constelación 16-QAM sin ecualización (izquierda) y con ecualización (derecha).....	91
7.9. Comparación de la constelación BPSK sin corrección en fase (izquierda) y después de la corrección en fase (derecha).....	92
7.10. Comparación de la constelación QPSK sin corrección en fase (izquierda) y después de la corrección en fase (derecha).....	92
7.11. Comparación de la constelación 16-QAM sin corrección en fase (izquierda) y después de la corrección en fase (derecha).....	93
7.12. Comparación de la constelación 64-QAM sin corrección en fase (izquierda) y después de la corrección en fase (derecha).....	93
7.13. Prueba de la etapa de demodulación.....	94
7.14. Antena VERT2450.....	96
7.15. Antena LP0965.....	97
7.16. Primer escenario para la prueba del enlace inalámbrico OFDM.....	97
7.17. Contenido del campo de datos de las tramas enviadas.....	98
7.18. Gráfica de BER vs SNR para modulación BPSK 1/2.....	100
7.19. Gráfica de BER vs SNR para modulación 16 QAM 1/2.....	101
7.20. Segundo escenario para la prueba del enlace inalámbrico OFDM.....	102



7.21. Distancia vs BER para el segundo escenario con modulaciones BPSK 1/2 y 16-QAM 1/2, antena LP0965.....	103
7.22. Distancia vs BER para el segundo escenario con modulaciones BPSK 1/2 y 16-QAM1/2, antena VERT 2450.....	104
7.23. Tercer escenario para la prueba del enlace inalámbrico OFDM.....	105
7.24. Distancia vs BER para el tercer escenario con modulaciones BPSK 1/2 y 16-QAM 1/2, antena LP0965	106



LISTA DE TABLAS

Tabla 3.1. Contenido del campo RATE dependiendo de la tasa de transmisión ...	37
Tabla 3.2. Parámetros dependientes de la tasa de transmisión.....	39
Tabla 3.3. Factor de normalización después de la modulación.....	43
Tabla 4.1. Características de la tarjeta madre.....	52
Tabla 5.1. Tipos de datos en LabVIEW.....	60
Tabla 7.1. Parámetros del sistema de transmisión.....	95
Tabla 7.2. Parámetros del sistema de recepción.....	95
Tabla 7.3. SNR vs BER para el esquema de modulación BPSK 1/2.....	99
Tabla 7.4. SNR vs BER para el esquema de modulación 16-QAM 1/2.....	100
Tabla 7.5. Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 2 con antena LP0965.....	103
Tabla 7.6. Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 2 con antena VERT 2450.....	104
Tabla 7.7. Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 3....	106



ACRÓNIMOS

A/D	Analógico – Digital
ADC	Analog to Digital Converter
ACK	Acknowledgement
AGC	Automatic Gain Control
AID	Association ID
AP	Access Point
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BSS	Basic Service Set
CCK	Complementary Code Keying
CFP	Contention-Free Period
CP	Cyclic Prefix
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
CTS	Clear to Send
CRC	Cyclic Redundancy Check
D/A	Digital – Analógico
DAC	Digital to Analog Converter
DDC	Digital Down Converter
DUC	Digital Up Converter
DBPSK	Differential Binary Phase Shift Keying
DCF	Distributed Coordination Function
DIFS	Distributed Coordination Function IFS
DS	Distribution System
DSP	Digital Signal Processor
DQPSK	Differential Quaternary Phase Shift Keying



DSSS	Direct Sequence Spread Spectrum
ESS	Extended Service Set
ERP	Extended Rate PHY
FCS	Frame Check Sequence
FDM	Frequency Division Multiplexing
FFT	Fast Fourier Transform
FHSS	Frequency Hopping Spread Spectrum
FPGA	Field Programmable Gate Array
GSM	Global System for Mobile Communications
IBSS	Independent Basic Service Set
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
IFS	Inter-Frame Space
ISI	Inter-Symbol Interference
ISM	Industrial Scientific Medical
IR	Infrared Light
LabVIEW	Laboratory Virtual Instrument Engineering Workbench
LLC	Logical Link Control
LTE	Long Term Evolution
MAC	Media Access Control
MIMO	Multiple Input Multiple Output
MPDU	MAC Layer Protocol Data Unit
NAV	Network Allocation Vector
OFDM	Orthogonal Frequency Division Multiplexing
OQPSK	Offset-Quaternary Phase Shift Keying
OSI	Open System Interconnection
PBCC	Packet Binary Convolutional Code
PCB	Printed Circuit Board
PCF	Point Coordination Function
PDR	Packet Delivery Ratio



PC	Personal Computer
PCIe	Peripheral Component Interconnect Express
PIFS	Point Coordination Function IFS
PLCP	Physical Layer Convergence Procedure
PLME	Physical Layer Management Entity
PMD	Physical Medium Dependent
PPDU	PLCP Protocol Data Unit
PPM	Pulse Position Modulation
PRR	Packet Received Ratio
PS	Power-Save
QAM	Quadrature Amplitude Modulation
QPSK	Quaternary Phase Shift Keying
RF	Radio Frecuencia
RTS	Request to Send
SDR	Software Defined Radio
SNR	Signal to Noise Radio
SR	Software Radio
TCP	Transmission Control Protocol
UMTS	Universal Mobile Telecommunications System
USRP	Universal Software Radio Peripheral
USB	Universal Serial Bus
VI	Virtual Instrument
VPSS	Video Processing Subsystem
Wi-Fi	Wireless Fidelity
WEP	Wireless Equivalent Privacy
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WRAN	Wireless Regional Area Network



CAPÍTULO I. INTRODUCCIÓN

1.1. Antecedentes

La constante necesidad del ser humano de permanecer siempre comunicado con el mundo que le rodea y el avance de la tecnología a pasos agigantados han permitido la evolución de los sistemas de comunicación.

En la actualidad, las comunicaciones inalámbricas juegan un papel fundamental en el día a día, pues permiten la transmisión de voz, datos y video, además de ofrecer movilidad al usuario.

Prueba de la importancia e impacto de las redes inalámbricas es el desarrollo de múltiples estándares y tecnologías, como los definidos por el IEEE (Institute of Electrical and Electronics Engineers) que contienen las especificaciones para implementación de redes inalámbricas con distintas áreas de cobertura.

Algunos estándares definidos por la IEEE son, por ejemplo, el estándar IEEE 802.11 para redes WLAN (Wireless Local Area Network), IEEE 802.15 para redes WPAN (Wireless Personal Area Network) que abarca las tecnologías Bluetooth y ZigBee, IEEE 802.16 para WMAN (Wireless Metropolitan Area Network) y la norma IEEE 802.22 para WRAN (Wireless Regional Area Network).

Otras tecnologías inalámbricas de gran importancia son las tecnologías de telefonía celular, por ejemplo GSM (Global System for Mobile Communications), UMTS (Universal Mobile Telecommunications System) o LTE (Long Term Evolution).



En particular, la especificación IEEE 802.11 es un estándar que define las capas física y de enlace de datos, específicamente la subcapa MAC (Media Access Control) para redes inalámbricas de área local o redes Wi-Fi.

El término Wi-Fi (Wireless Fidelity) es el nombre de la certificación que es otorgada por la *Wi-Fi Alliance* para garantizar la compatibilidad entre dispositivos que cumplan con el estándar IEEE 802.11.

Actualmente la norma 802.11 cuenta con distintas versiones, entre las que destacan las versiones a,b,g,n,p,ac y ad que definen distintos tipos de modulación, ancho de banda y frecuencia de operación.

Uno de los retos más importantes en los últimos años en el área de las comunicaciones inalámbricas es la integración de diferentes estándares y protocolos en un solo equipo de comunicación y es por esta razón por la que la plataforma SDR ha tomado cada vez más relevancia en el campo de las comunicaciones, puesto que es una tecnología reconfigurable, es decir, ofrece la flexibilidad de definir la capa física y las características del sistema de comunicación vía software.

1.2 Definición del Problema

Cuando se habla de comunicaciones inalámbricas hay una gran cantidad de tecnologías y protocolos, los cuales definen las características de un sistema de comunicaciones en particular. Tradicionalmente, dichas tecnologías eran implementadas en sistemas de comunicación basados solamente en hardware de propósito específico y una vez construidos, su funcionalidad permanecía fija y no podía ser modificada fácilmente.



Con la llegada de potentes DSPs (Digital Signal Processors) y FPGAs (*Field Programmable Gate Arrays*) es ahora posible definir algunas características del sistema vía software como el tipo de modulación, ganancia, ancho de banda y frecuencia central, las cuales anteriormente dependían exclusivamente de los componentes de hardware del sistema, lo que ha dado lugar al campo de los Radios Definidos por Software (SDR), cuyas características y funciones de capa física son implementados en una plataforma programable y reconfigurable.

La ventaja que ofrece la plataforma SDR es la capacidad de implementar distintos tipos de tecnologías en un mismo dispositivo, disminuyendo así el costo de implementación.

Debido a que la tecnología SDR se encuentra en pleno desarrollo, es necesario el estudio e implementación de distintas tecnologías de comunicación inalámbrica para poder evaluar plenamente el desempeño de una plataforma SDR.

La plataforma SDR en el campo de la investigación permite explorar distintas aproximaciones para la implementación de un sistema a nivel de capa física y así poder analizar su comportamiento hasta encontrar la implementación óptima para obtener un sistema de comunicación de bajo costo y eficiente.

Debido a la flexibilidad de SDR, se decidió implementar la capa física de un receptor basado en el estándar IEEE 802.11g bajo ésta plataforma. El estándar IEEE 802.11g es una modificación del estándar original IEEE 802.11 el cual es un protocolo de comunicaciones para redes inalámbricas de área local, de gran importancia en la vida cotidiana, pues brinda acceso a la red a los dispositivos sin importar su ubicación y movilidad.

La implementación de las funciones de capa física de la norma IEEE 802.11g en un SDR ofrece la posibilidad de implementar y evaluar distintos algoritmos que ayuden a obtener un receptor de bajo costo y una comunicación con un bajo índice de BER (Bit Error Rate).



Una de las herramientas más conocidas para implementar SDR y en el cual se implementó el presente proyecto es el software LabVIEW, el cual es una plataforma para el diseño de sistemas por medio de un lenguaje de programación gráfico e intuitivo.

1.3 Objetivos

El principal objetivo para la elaboración de éste trabajo es el de implementar la capa física de un receptor OFDM (Orthogonal Frequency Division Multiplexing) tomando como base el estándar IEEE 802.11g en un Radio Definido por Software programado en LabVIEW.

El propósito de éste trabajo de tesis es tener un acercamiento práctico, por medio de la plataforma SDR, al diseño de sistemas de comunicación inalámbrica y así evaluar su desempeño y viabilidad de ser implementados en ambientes de trabajo reales.

Se deberán diseñar y desarrollar los bloques necesarios para el procesamiento de la señal tales como la corrección en frecuencia, la alineación del símbolo, estimación del canal, demodulación y decodificación en el lenguaje de programación gráfico de LabVIEW, además se validarán los algoritmos usados para dichos bloques.

Se verificará el funcionamiento del receptor mediante 3 distintos escenarios (dos escenarios en interiores y uno en exteriores) con la ayuda de un transmisor OFDM construido bajo la misma plataforma (basado en el estándar IEEE802.11g con un SDR programado en LabVIEW), el cual será el encargado del envío de tramas de datos, dichas tramas serán captadas con el receptor OFDM para medir parámetros del sistema como BER vs Distancia y BER vs SNR, además se obtendrá la tasa de transferencia para así lograr caracterizar el sistema.



1.4 Metodología

Para cumplir con los objetivos de ésta tesis, descritos anteriormente, se llevará a cabo su desarrollo en dos etapas, el estudio teórico y el desarrollo práctico.

En el estudio teórico, el objetivo es revisar los conceptos teóricos que son los pilares de éste trabajo de tesis.

Estudio del Estándar IEEE 802.11

- Investigación del estado actual de las redes Wi-Fi.
- Investigación y análisis de las funciones realizadas a nivel de capa PHY.
- Investigación y análisis de los algoritmos necesarios para la recepción de tramas OFDM para el estándar IEEE 802.11.

Estudio de la plataforma SDR

- Investigación del estado actual de la tecnología SDR.
- Investigación de la arquitectura y funcionamiento del SDR.

Estudio del software para la programación del sistema SDR

- Investigación del funcionamiento y manejo del software en general.
- Investigación del manejo de la plataforma SDR con el software.

En el desarrollo práctico la finalidad es obtener un modelo funcional de un receptor OFDM reconfigurable por software, por lo que la metodología a seguir es:

- Diseño del esquema de recepción OFDM.
- Programación en la plataforma SDR de cada algoritmo.
- Verificación y validación de los algoritmos del receptor OFDM



1.5 Contribuciones y Relevancia

Con la realización del estado del arte del presente trabajo de tesis se pretende dar un panorama general de la plataforma SDR, su desarrollo y el estado de la tecnología a la fecha.

Una de las contribuciones del presente trabajo de investigación es obtener un algoritmo eficiente para la recepción de señales OFDM de acuerdo al estándar IEEE 802.11g.

Se obtendrá el primer receptor OFDM trabajando con el estándar IEEE 802.11g programado en LabVIEW.

La implementación de éste sistema permitirá el desarrollo de futuros estudios de capa física y de igual forma, facilitará la implementación de las capas superiores del modelo OSI (Open System Interconnection).

Al final de éste trabajo se tendrá un estudio del desempeño de un sistema de comunicación (Transmisor / Receptor) en equipos SDR bajo el estándar 802.11g y finalmente se analizará la viabilidad de su implementación en ambientes reales por medio de mediciones de BER y velocidad de transmisión.

1.6 Estructura de la tesis

El presente trabajo se encuentra dividido en 7 capítulos. En el capítulo 2 se brinda una introducción a las redes inalámbricas Wi-Fi y su estado actual, también se aborda un estudio de los trabajos relacionados con éste proyecto, se presenta una recopilación de los modelos existentes para receptores OFDM en dicha plataforma.

En el capítulo 3 se da una descripción del estándar IEEE 802.11, en donde se incluyen los detalles más importantes de la capa física y la capa de acceso al medio, los cuales son necesarios para entender su funcionamiento y finalmente lograr su implementación vía software.



En el capítulo 4 se ofrece una descripción detallada de la plataforma SDR, sus componentes principales y se brinda la descripción del equipo USRP (Universal Software Radio Peripheral) usado como receptor OFDM, el USRP N210 de Ettus Research. Se describe su arquitectura, sus características más importantes y se mencionan las plataformas en las que puede ser programado.

El capítulo 5 contiene una introducción al lenguaje de programación LabVIEW usado en éste proyecto, se estudian sus características principales, su método de programación, así como las funciones necesarias para programar el SDR con éste software.

En el capítulo 6 se presenta la descripción de la propuesta de este trabajo, se incluye uno a uno los algoritmos, fundamentos matemáticos y códigos en LabVIEW usados para lograr la recepción y decodificación de los símbolos OFDM: corrección en frecuencia, alineación del símbolo, estimación del canal, ecualización del símbolo, demodulación, deintercalamiento, decodificación y derandomización.

Finalmente el objetivo del capítulo 7 es el de presentar los resultados obtenidos, se presentan las pruebas realizadas para la comprobación de los algoritmos implementados. Además se presentan algunas pruebas de campo, se presenta un análisis de las pruebas realizadas, como es el caso de la medición del BER en función del SNR (Signal to Noise Ratio) y de la distancia, por tipo de modulación y la tasa de transferencia alcanzada.



CAPÍTULO II. ESTADO DEL ARTE

Para lograr la elaboración de éste proyecto, es de vital importancia conocer el nivel de desarrollo actual de las áreas relacionadas con éste trabajo. Por un lado se encuentra el desarrollo de las redes inalámbricas de área local o redes WLAN, las cuales han hecho posible que, ya sea en el hogar o en una oficina, múltiples equipos compartan recursos vía inalámbrica (conexión a internet, impresoras, archivos de datos, audio o video, etc.) a cada vez mayor velocidad. Por otro lado, es fundamental conocer algunos de los proyectos realizados bajo la plataforma SDR, con el objetivo de identificar los avances más recientes en el campo de la tecnología SDR y adquirir un mayor dominio del tema.

El presente capítulo está dividido en 2 secciones, la primera sección da una introducción a las redes Wi-Fi, sus características principales y presenta las versiones existentes del estándar IEEE 802.11. La segunda contiene una recopilación de los trabajos relevantes para el presente proyecto de investigación.

2.1 Redes Wi-Fi

2.1.1 Introducción

El término Wi-Fi se refiere a la certificación otorgada por la Wi-Fi Alliance para garantizar la interoperabilidad de los equipos que forman una red inalámbrica de área local bajo el estándar IEEE 802.11. Actualmente la tecnología Wireless LAN es un componente crucial de las redes de computadoras pues ofrece movilidad y servicios de redes donde la instalación de una red cableada no es viable.



A finales de la década de 1980 el grupo de trabajo 802 del Instituto de Ingenieros Eléctricos y Electrónicos responsable del desarrollo de los estándares para redes de área local comenzó con el desarrollo del estándar para redes WLAN.

La primera versión del estándar IEEE 802.11 fue aprobada por la IEEE el 26 de Junio de 1997 y publicada posteriormente en Noviembre del mismo año [1].

2.1.2 Ventajas de las redes WLAN

El constante crecimiento y desarrollo de las redes WLAN se debe a la creciente necesidad de encontrar formas para la implementación de redes cada vez más veloces, eficientes y a un menor costo. En general las ventajas que las redes inalámbricas de área local ofrecen son:

- **Movilidad:** Esta característica brinda flexibilidad a los usuarios de una red WLAN al permitir que se desplacen físicamente mientras se encuentran utilizando un dispositivo portable conectado a la red, lo que conlleva una mayor productividad y eficiencia.
- **Reducción de costos a largo plazo:** La instalación de redes inalámbricas tiene una reducción de costos tangible cuando se implementa en lugares donde sería difícil realizar una red cableada.
- **Mayor fiabilidad:** Un problema inherente con las redes cableadas es el tiempo de inactividad debido a la falla del alambrado, lo que impide que el usuario tenga acceso a los recursos de la red; ya sea, por ejemplo, por la corrosión de los conductores metálicos provocada por la humedad o por el daño en los conectores debido al mal uso o al uso continuo de éstos, etc.



- **Escalabilidad y menor tiempo de instalación:** Una vez implementada una red inalámbrica de área local es posible incrementar el número de nodos en la red fácilmente, a diferencia de las redes cableadas en donde el proceso de instalación puede ser lento y costoso.

2.1.3 La especificación IEEE 802.11

La familia IEEE 802.11 está conformada por distintas versiones que pueden ser usadas al implementar redes WLAN. A continuación se presentan los estándares más comunes y se destacan sus características más importantes.

- **IEEE 802.11 [2]:** La versión original del estándar IEEE 802.11 fue publicada en 1997 y actualmente es obsoleta. Define 3 tecnologías de capa física: Infrarrojo y dos formas de modulación de espectro expandido: FHSS (Frequency Hopping Spread Spectrum) y DSSS (Direct Sequence Spread Spectrum).

Las tasas de transferencia definidas en el estándar original son de 1 Mbps y 2 Mbps en la banda ISM (Industrial Scientific Medical) de 2.4 GHz para las tecnologías FHSS y DSSS, el ancho de banda del canal es de 20 MHz.

- **IEEE 802.11a [3]:** Publicado en 1999, el estándar IEEE 802.11a opera en la banda de los 5 GHz que al estar menos saturada, existe menor interferencia con otros dispositivos, puede alcanzar una velocidad máxima de transferencia de 54 Mbps con modulación OFDM y opera con un ancho de banda de 20 MHz.

Debido a que su frecuencia de operación es mayor en comparación con la banda de 2.4 GHz, su área de cobertura es menor y es menos eficiente al penetrar estructuras de construcción.

Los dispositivos que operan bajo este estándar no son compatibles con las especificaciones 802.11b ni 802.11g.



- **IEEE 802.11b** [4]: El estándar IEEE 802.11b fue publicado en 1999 y opera en la banda de 2.4 GHz. Ofrece tasas de transferencia de hasta 11 Mbps, con un ancho de banda de 20 MHz y trabaja con la modulación DSSS con CCK (Complementary Code Keying).
- **IEEE 802.11g** [5]: Publicado en 2003, opera en la banda de 2.4 GHz con ancho de banda del canal de 20 MHz y ofrece velocidades de transmisión de hasta 54 Mbps. Utiliza modulación DSSS y OFDM.
Es compatible con dispositivos que trabajen con la norma IEEE 802.11b.
- **IEEE 802.11n** [6]: Este estándar fue dado a conocer en 2009, opera en las bandas de 2.4 y 5 GHz con canales de 20 y 40 MHz, las tasas de datos pueden ser de 150 Mbps y 600 Mbps con una distancia de hasta 70m.
Para alcanzar las velocidades más altas, los APs (Access Points) y clientes requieren múltiples antenas (hasta 4) usando la tecnología MIMO (Multiple Input Multiple Output). MIMO usa múltiples antenas tanto en la transmisión como en la recepción para mejorar el desempeño de la comunicación.
Es compatible con dispositivos 802.11a/b/g, sin embargo cuando dispositivos con distintos estándares se encuentran presentes en la red, esto limita las tasas de transferencia alcanzadas.
- **IEEE 802.11ac** [7]: Fue publicado en el año 2013, este estándar opera en la banda de 5 GHz, con velocidades de transferencia de datos de hasta 450 Mb/s y 1.3 Gb/s.



Usa la tecnología MIMO para mejorar el desempeño de la red y aumentar las velocidades de transmisión, puede soportar hasta 8 antenas. Los anchos de banda del canal pueden ser de 20, 40, 80 y 160 MHz. Es compatible con dispositivos que operan bajo los estándares 802.11a/n

- **IEEE 802.11ad [8]:** Es también conocido como 'WiGig'. Utiliza tres bandas de operación: 2.4 GHz, 5 GHz y 60 GHz con velocidades de transmisión teóricas de hasta 7 Gb/s.

Es compatible con dispositivos que operan bajo los estándares 802.11a/b/g/n/ac.

2.2 Trabajo Relacionado

A pesar de que el término SDR fue introducido hace ya más de dos décadas [9], el campo de los radios definidos por software se encuentra en pleno desarrollo, el objetivo de esta sección es brindar un panorama de los trabajos relacionados al presente proyecto que se han realizado hasta la fecha.

En el año 2006, Di Stefano [10] propuso el diseño de un prototipo de SDR, en donde la etapa de RF (Radio Frecuencia) era fija, trabajando solamente en la banda ISM de 2.4 GHz, contaba con un FPGA para realizar el procesamiento en banda base y para realizar mediciones, una antena de microcinta triangular omnidireccional diseñada para trabajar a 2.4 GHz, dos convertidores A/D de 6 bits operando a una velocidad de 22 Msps, cada una para las señales de fase y cuadratura , una fuente de voltaje y una computadora como la encargada del control y la configuración del sistema.



El prototipo de Di Stefano fue diseñado para recibir mensajes de redes Wi-Fi y ZigBee (estándar IEEE 802.11 y 802.15.4), con anchos de banda de 22 MHz y 2 MHz respectivamente. Para el estándar IEEE 802.11 se usó modulación DBPSK (Differential Binary Phase Shift Keying) para el preámbulo de cada trama y DQPSK (Differential Quaternary Phase Shift Keying) para el campo de datos, trabajando con la tecnología DSSS. Para la norma IEEE 802.15.4 se usó la modulación OQPSK (Offset – Quaternary Phase Shift Keying) y de igual forma la tecnología DSSS.

Tachwali [11] en 2008 realizó la implementación de un transmisor y receptor basado en la plataforma SDR para un rango de frecuencia de 200 – 900 MHz y para un esquema de modulación BPSK (Binary Phase Shift Keying). El prototipo de Tachwali fue diseñado usando una arquitectura híbrida, es decir, usando un FPGA y un DSP.

En el FPGA se implementó la etapa de filtrado digital, modulador/demodulador, interpolador, decimador, el conversor A/D (de 14 bits a una velocidad de 125 Msps), el convertidor D/A (de 16 bits a una velocidad de 500 Msps), la etapa de conversión a frecuencia intermedia y posteriormente a radio frecuencia para la transmisión y de conversión a banda base para recepción.

El DSP fue usado para la etapa de sincronización y la función AGC (Automatic Gain Control). Finalmente se usó un subsistema VPSS (Video Processing Subsystem) para hacer posible la comunicación entre el FPGA y el DSP.

La implementación del prototipo de Tachwali se realizó en modo bucle, en donde la salida del transmisor se conectó directamente a la entrada del receptor y se obtuvieron las estadísticas correspondientes a la utilización de recursos del FPGA.

Un año más tarde, en el 2009, Marwanto [12] realizó un estudio para la implementación de un sistema OFDM usando el software GNU Radio y la plataforma USRP. A diferencia de los sistemas mencionados previamente [10] [11], en donde cada componente en hardware que conformaba el SDR fue previamente seleccionado, el estudio de Marwanto se basó en una plataforma de SDR comercial y de bajo costo, el USRP.



El sistema OFDM en [12] constaba de dos USRP (transmisor y receptor) con tarjetas hija modelo RFX2400 el cual opera en un rango de frecuencias entre 2.3 GHz y 2.9 GHz. La distancia entre ambos USRP en la implementación del sistema era de 66 cm en un ambiente cerrado, trabajando a una frecuencia central de 2.5 GHz y con esquemas de modulación BPSK y QPSK (Quaternary Phase Shift Keying) con 256 y 512 portadoras. El objetivo del sistema fue determinar la relación entre la tasa de paquetes recibidos o PRR (Packet Received Ratio) y la potencia de transmisión.

En el año 2010, Gutiérrez – Agullo [13] desarrolló las funciones de la capa MAC (Media Access Control) del estándar IEEE 802.11 en lenguaje python para ser usada bajo la plataforma USRP. Las funciones de la capa de control de acceso al medio fueron implementadas en base a una máquina de estados.

El sistema propuesto en [13] era capaz de emular los mensajes de transmisión de la capa de control de acceso al medio como son las tramas de datos, RTS (Request to Send), CTS (Clear to Send), ACK (Acknowledgement) y beacons, usando como base la capa física del sistema transmisor OFDM propuesto por Fuxjäger [14], aunque con un número de símbolos OFDM variable.

Como parte de las funciones de la capa MAC, se implementó el método de acceso al medio CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance) de acuerdo a la norma 802.11 para determinar si el canal se encuentra disponible para iniciar la transmisión, el algoritmo de *back-off* para evitar que existan colisiones si varios nodos tratan de transmitir al mismo tiempo y el mecanismo RTS/CTS para evitar el problema de la terminal oculta y la terminal expuesta.

El trabajo de Gutiérrez – Agullo comprendió la implementación de un transmisor funcional en cuanto a capa PHY y a capa MAC, el cual fue puesto a prueba usando un analizador de protocolos o *sniffer* para confirmar que los paquetes generados por el transmisor cumplieran con lo establecido por el estándar.



En cuanto a la parte de recepción, la capa MAC fue desarrollada de igual forma, aunque el enlace inalámbrico no fue puesto a prueba debido a que no se tenía desarrollada la capa PHY en la recepción.

El primer receptor OFDM basado en la plataforma SDR fue diseñado en el año 2013 por Bloessl [15]. Dicho trabajo consistió en la implementación de un receptor OFDM en el software GNU Radio operando con el hardware USRP y basado en la capa PHY del estándar IEEE 802.11 a/g/p. De igual forma que en [13] el transmisor estaba basado en el transmisor propuesto por Fuxjäger [14].

El sistema presentado en [15], constaba de dos equipos USRP con tarjetas hija modelo XCVR2450 controlados con 2 PCs, ambos trabajando en la banda de 2.4 GHz y 5GHz, con un ancho de banda de 20 MHz y 10 MHz, operando con modulaciones BPSK y QPSK. En general, la parte de recepción se realizó en dos etapas principales: la primera parte, la detección del frame y la segunda parte, la decodificación del mensaje, la cual está conformada por la corrección en frecuencia y fase, la alineación del símbolo OFDM, estimación del canal y la decodificación del campo de señal y del campo de datos.

En el trabajo realizado por Bloessl se realizaron algunas pruebas de interoperabilidad al recibir paquetes enviados desde tarjetas de red comerciales, en donde se lograron recibir tramas como beacons, frames de datos y de control como RTS, aunque no tramas CTS. También se realizaron pruebas para determinar el comportamiento del sistema, se enviaron 5 paquetes por segundo de 63 bytes cada uno con un esquema de modulación BPSK a una frecuencia de 5GHz y se obtuvo el PDR (Packet Delivery Ratio) en función de la potencia de transmisión.

Ya para el año 2012, National Instruments incluyó como parte del software LabVIEW el soporte para la plataforma USRP. Por tal motivo, en el año 2013, Shanin [16] desarrolló un sistema de recepción para la caracterización del canal para el estándar 802.11g usando un equipo USRP.



El objetivo del trabajo de Shanin fue medir características de los canales usados en el estándar IEEE 802.11g tal como: el ciclo útil (el porcentaje de tiempo que el canal se encuentra ocupado en un periodo de tiempo determinado), tasa de llegada de los paquetes, tiempo entre paquetes recibidos y la distribución de los nodos en la red (la distancia aproximada de los nodos en relación con el USRP) usando la plataforma SDR y el software LabVIEW con el objetivo de determinar las condiciones para lograr una comunicación inalámbrica más confiable.

Finalmente un año después, en el 2014, Lei [17] presentó la implementación de un enlace inalámbrico OFDM usando LabVIEW y el USRP.

En primer lugar se generó una secuencia pseudo-aleatoria, la cual fue modulada en LabVIEW y posteriormente transmitida al USRP para luego ser enviada a una frecuencia central de 915 MHz y recibida con otro equipo USRP. Los esquemas de modulación implementados por el sistema de Lei fueron BPSK, QPSK, 16 QAM (Quadrature Amplitude Modulation) y 64 QAM con un tamaño de FFT de 256, un ancho de banda entre 1.5 MHz y 2 MHz y una longitud de ciclo prefijo de 1/32, en total cada mensaje constaba de 2044 simbolos y como resultados de dicho trabajo se obtuvo una comparación de la constelación de transmisión y de recepción para las modulaciones QPSK y 16 QAM, además del espectro de la señal recibida.

De los trabajos relacionados presentados en esta sección, se puede observar el gran avance que se ha logrado en el campo de los SDR, partiendo de prototipos modulares propuestos en laboratorio hasta llegar a plataformas SDR comerciales como es el caso del USRP. Por otro lado, también es necesario destacar que hasta la fecha no se tiene conocimiento de un receptor OFDM en SDR con las características de capa física del estándar IEEE 802.11 programado en LabVIEW, ya que a pesar que el trabajo de Lei [17] fue realizado en LabVIEW, no presenta las funciones tales como sincronización, estimación de canal, scrambler, interleaver, codificación convolucional y además no se presentó ningún análisis en cuanto a BER. Por otro lado, el prototipo propuesto por Bloessl [15] que sí presenta dichas funciones, se realizó en otro software (GNU Radio) pero no se realizó para esquemas de modulación 16-QAM y 64-QAM.



CAPÍTULO III. EL ESTÁNDAR IEEE 802.11

El estándar IEEE 802.11 es una especificación desarrollada por la IEEE en 1997, el cual define las características de una red WLAN, específicamente las funciones a nivel de capa PHY y de capa MAC de los dispositivos que conforman una red inalámbrica de área local.

El objetivo de éste capítulo es brindar una revisión a detalle de las características principales del estándar IEEE 802.11, para entender su funcionamiento y así realizar su implementación.

En éste capítulo se presenta una descripción de la arquitectura de las redes Wi-Fi, así como de las capas definidas por el estándar 802.11: la capa de acceso al medio (MAC) y la capa física (PHY), específicamente para la modulación OFDM de la versión 'g' de dicho estándar, resaltando sus características y ventajas por las cuales se eligió éste esquema de modulación como la base de éste proyecto.

3.1 Introducción

3.1.1 La familia de estándares IEEE 802

El estándar IEEE 802.11 forma parte a su vez de la familia IEEE 802, el cual es un comité de estándares para redes de área local y de área metropolitana, además es uno de los grupos de trabajo más importantes formado por el Instituto de Ingenieros Eléctricos y Electrónicos en Febrero de 1980.



La familia de estándares IEEE 802 se centra en las dos capas más bajas del modelo OSI [1], la capa física y la capa de enlace de datos. La familia IEEE 802 incluye un gran número de especificaciones, algunos de los cuales se muestran en la Figura 3.1.

En la familia 802 la capa de control de acceso al medio (MAC) y física (PHY) se encuentran definidas por separado de la capa de control de enlace lógico (LLC) debido a la interdependencia de la capa MAC con el medio físico y la topología de la red.

IEEE 802.2 Control de Enlace Lógico (LLC)				LLC	OSI Capa 2 Enlace de Datos
IEEE 802.3 CSMA/CD	IEEE 802.4 Token Bus	IEEE 802.5 Token Ring	IEEE 802.11 Wi-Fi	MAC	OSI Capa 1 Física
				PHY	

Figura 3.1.- Familia de Estándares IEEE 802 y su relación con el modelo OSI

Como se puede apreciar en la figura 3.1, el estándar IEEE 802.11 es un protocolo que describe la capa MAC y PHY para redes inalámbricas de área local, también conocidas como redes Wi-Fi. El estándar IEEE 802.11 se apoya en la especificación 802.2 para la capa de control de enlace lógico, el cual es común para las redes LAN.

3.1.2 La topología de las redes IEEE 802.11

Los componentes que forman una red WLAN definidos en el estándar IEEE 802.11 se muestran en la Figura 3.2 [18].

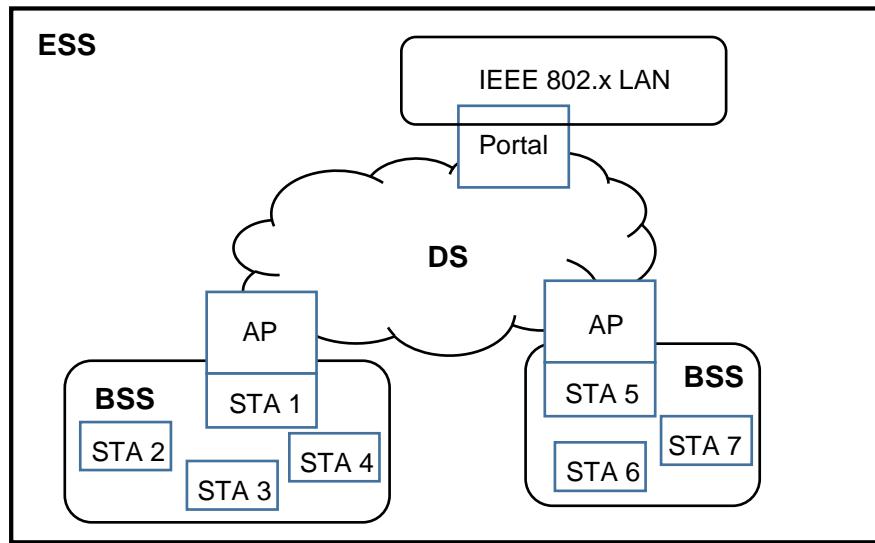


Figura 3.2 .- Arquitectura del estándar IEEE 802.11

Una **estación** (STA) representa cualquier dispositivo que contiene la funcionalidad del protocolo 802.11, en otras palabras la capa MAC, la capa PHY y una interface al medio inalámbrico. El elemento básico de una red LAN inalámbrica es el llamado conjunto de servicios básico o **BSS** (Basic Service Set), el cual contiene un cierto número de estaciones trabajando con el mismo protocolo MAC y compitiendo por el acceso al mismo medio inalámbrico.

Un BSS puede estar aislado o puede estar conectado a un **sistema de distribución** (DS) o red troncal por medio de un **punto de acceso** (AP).

Cuando todas las estaciones móviles en un BSS no tienen conexión con otros BSS, es decir, cuando el BSS es independiente, entonces el BSS es llamado **IBSS** (Independent Basic Service Set). A este tipo de red se le conoce como red ad-hoc y por lo general se usa para redes con áreas de cobertura pequeña como una habitación o un nivel de una construcción.



Un conjunto de servicios extendido o **ESS** (Extended Service Set) consiste de dos o más BSS interconectados por medio de un sistema de distribución. Por lo general el DS se trata de una red LAN troncal. En los BSS que conforman el ESS, las estaciones no se comunican directamente entre ellas, en cambio, los paquetes de datos provenientes de la estación origen son enviados primero a un AP y del AP son enviados a la estación destino. Finalmente, un **portal** permite la comunicación entre redes inalámbricas IEEE 802.11 con una red LAN.

3.2 Capa MAC del estándar IEEE 802.11

En general la capa de control de acceso al medio debe cumplir dos funciones principales:

- Entrega de datos confiable
- Control de acceso al medio

3.2.1 Entrega de datos confiable

Como en cualquier red inalámbrica, una red WLAN es susceptible al ruido, interferencia y otros efectos de propagación, lo cual conlleva la pérdida de un significante número de paquetes.

A pesar de contar con códigos correctores de errores, un número de tramas MAC pueden no ser recibidas correctamente, lo cual puede ser resuelto por medio de mecanismos implementados en capas superiores, por ejemplo, el protocolo TCP (Transmission Control Protocol) en la capa de transporte. Sin embargo, los temporizadores usados para las retransmisiones en las capas superiores pueden ser del orden de segundos, es por eso que lidiar con errores a nivel de capa MAC es más eficiente.



Cuando una estación recibe tramas de datos de otra estación, la estación destino envía un mensaje de confirmación (ACK) como respuesta a la estación origen. Si la estación fuente no recibe el mensaje ACK en un periodo de tiempo corto, ya sea debido a que la trama de datos, enviada originalmente, se perdió o porque la confirmación se dañó, la estación fuente retransmitirá el mensaje.

Para mejorar la confiabilidad del enlace inalámbrico es posible usar un mecanismo de comunicación que implica la transmisión de cuatro mensajes. En primer lugar, la estación transmisora envía un frame RTS para informar a la estación destino que se desea enviar información, la estación destino responde con un mensaje CTS para avisar que está lista para recibir. Los mensajes RTS y CTS sirven a la vez para avisar a las otras estaciones que un intercambio de información se llevará acabo. Después de recibido el mensaje CTS, la estación transmisora envía las tramas de datos para que finalmente la estación destino confirme la recepción del mensaje con un ACK.

3.2.2 Control de Acceso al Medio

Antes de transmitir una trama de datos, la capa MAC debe ganar acceso a la red usando uno de los siguientes modos:

Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA):

Se trata de un protocolo similar al presentado en el estándar IEEE 802.3 de Ethernet. La especificación 802.11 se refiere a este mecanismo como la función de coordinación distribuida o DCF (Distributed Coordination Function), la cual permite que distintas estaciones y APs entren en contención para el uso del medio inalámbrico.

Si una estación tiene una trama MAC a transmitir, la estación debe sensar el medio. Si el medio se encuentra libre, la estación puede transmitir, de otra forma la estación debe esperar a que la transmisión actual termine.



El sensado del canal se lleva acabo tanto a un nivel físico como virtual. El sensado virtual se basa en la información contenida en los mensajes recibidos, específicamente en el campo de duración de la trama, esa información se almacena en el NAV (Network Allocation Vector) y una vez que el tiempo indicado en éste campo ha transcurrido, la estación puede transmitir siempre y cuando el sensado físico del medio indique un canal desocupado.

En general el algoritmo CSMA/CA se puede describir de la siguiente forma:

- 1.- Una estación con un frame a transmitir sensa el medio. Si el medio está desocupado, espera un tiempo conocido como IFS (Inter-Frame Space) y vuelve a escuchar el medio. Si el medio continúa libre, la estación transmite inmediatamente.
- 2.- Si el medio resulta ocupado la estación pospone la transmisión y continúa monitoreando el medio hasta que la transmisión actual termine.
- 3.- Una vez que la transmisión ha terminado, la estación espera otro tiempo IFS.

Si el medio permanece libre por este periodo, entonces la estación ejecuta un algoritmo llamado *Exponential Backoff*, en donde la estación espera un tiempo aleatorio con distribución uniforme para volver a sensar el medio, con el objetivo de reducir la probabilidad de colisión entre dos estaciones que requieran realizar una transmisión. Si el medio sigue libre, la estación realiza la transmisión.

Si durante el tiempo aleatorio el medio fue ocupado, el temporizador de la estación se detiene y reinicia una vez que el medio se desocupe.

- 4.- Si la transmisión no fue exitosa, es decir, si no hay una confirmación de la estación destino, entonces se asume que ocurrió una colisión. Después de cada colisión el valor medio del retardo aleatorio se duplica hasta llegar a un valor máximo.

El tiempo IFS usado en CSMA/CA puede tener distintos valores dependiendo del nivel de prioridad del mensaje a transmitir, como lo muestra la Figura 3.3.

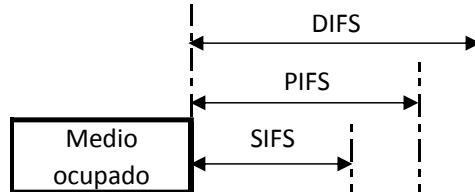


Figura 3.3 .. Distintas duraciones del intervalo IFS

SIFS (Short IFS): Es la versión más corta del tiempo IFS, por lo que se usa para brindar el mayor nivel de prioridad, permitiéndole a ciertas tramas acceder al medio antes que otras, entre los tipos de tramas que usan el intervalo SIFS están: ACK, CTS y la segunda trama de datos y posteriores pertenecientes al mismo ciclo de transmisión.

PIFS (Point Coordination Function IFS): Es el intervalo que las estaciones que operen con la función PCF (Point Coordination Function) usan para tener mayor prioridad respecto a los frames enviados bajo DCF.

DIFS (DCF IFS): Todas las estaciones operando de acuerdo a la función DCF usan el intervalo DIFS para transmitir frames de datos y de gestión.

Point Coordination Function (PCF)

La función de coordinación puntual o PCF es un método de acceso alternativo a DCF que permite la transmisión de frames sin contención por un periodo de tiempo llamado CFP (Contention-Free Period) y es útil para el envío de información en donde el tiempo es un factor crítico.

En este modo de operación existe un coordinador (el AP), el cual transmite y permite transmitir tramas a ciertas estaciones (configuradas para trabajar con PCF) en un periodo IFS más corto de tiempo (PIFS) en comparación con aquellas que usan DCF (DIFS).



3.2.3 Formato de la trama MAC

Cada frame está compuesto por tres partes fundamentales:

- *Encabezado MAC*: El cual incluye los campos de control, duración, direcciones MAC e información de control de secuencia.
- *Cuerpo de la trama*: Se trata de los datos útiles provenientes de capas superiores y tiene longitud variable.
- *Campo FCS (Frame Check Sequence)*: Contiene un código de redundancia cíclica o CRC (Cyclic Redundancy Check) de 32 bits.

El formato general de la trama de capa MAC para el estándar IEEE 802.11 se aprecia en la Figura 3.4, se muestra cada uno de los campos que conforman la trama, así como la longitud en bytes de cada uno.

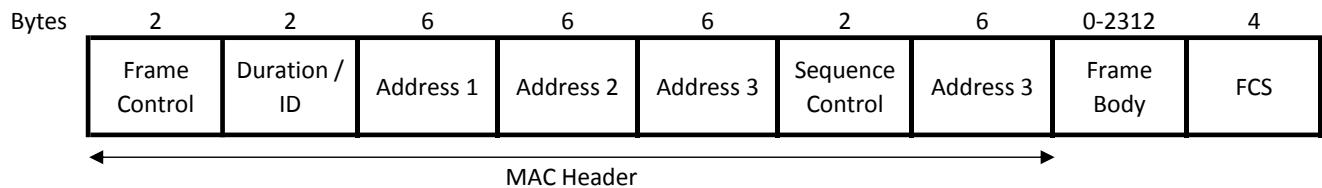


Figura 3.4 .- Formato general de una trama MAC

Como se puede apreciar en la Figura 3.4, los campos que componen un frame son:

- *Frame Control*
- *Duration / ID*: En tramas PS (Power-Save) para dispositivos con limitaciones de potencia contiene el identificador AID (Association ID) de 14 bits de longitud, los otros dos bits restantes en '1'. Para los otros tipos de frame, este campo contiene la duración del periodo que se ha reservado el medio para una estación.



- *Address 1-4:* Contiene las direcciones de 48 bits donde se especifica la dirección de la estación transmisora, estación receptora, el AP de origen y el AP de destino.
- *Sequence control:* Contiene el número de secuencia y el número de fragmento del frame que está siendo enviado.
- *Cuerpo de la trama:* Es la carga de datos útiles proveniente de protocolos de capas superiores y varía según el tipo de trama que se desee enviar.
- *FCS:* Verifica la integridad de la trama enviada mediante un código de redundancia cíclica de 32 bits.

El campo de control (*Frame control*) de 16 bits esta a su vez formado por otros subcampos, los cuales se muestran en la Figura 3.5.

B0	B1	B2	B3	B4	B7	B8	B9	B10	B11	B12	B13	B14	B15
Protocol Version	Type	Subtype		To DS	From DS	More Frag	Retry	Pwr Mgt	More Data	WEP	Order		
Bits	2	2	4	1	1	1	1	1	1	1	1	1	1

Figura 3.5 .- Estructura del campo Frame Control

Los subcampos que conforman el campo de control son:

- *Protocol Version:* Indica la versión del estándar 802.11 y ambos bits son '0'.
- *Type / Subtype:* Identifican la función del frame, el campo *Type* indica si es un frame de datos, de control o de gestión, adicionalmente se debe especificar el subtipo de frame.



- *To DS / From DS*: Indica si la trama se envía o es recibida del sistema de distribución DS.
- *More Fragments*: Es puesto a 1 si el paquete proveniente de capas superiores ha sido fragmentado.
- *Retry*: Si el frame es una retransmisión este campo se coloca en '1'.
- *Power Management*: Activa el modo de ahorro de energía.
- *More Data*: Indica si una estación tiene más tramas que transmitir para una estación destino específica.
- *WEP Field*: Indica si la trama ha sido procesada con el algoritmo de encriptación WEP (Wireless Equivalent Privacy).
- *Order Field*: Es activado cuando es necesario el envío ordenado de paquetes.

Existen tres tipos de tramas MAC definidas en el estándar IEEE 802.11, éstos son:

- Management: Las tramas de gestión tienen como objetivo establecer la comunicación entre las estaciones y el AP. Por lo tanto en este tipo de frames se incluyen los mensajes de asociación, autenticación, beacons, etc.
- Control: Una vez que una estación se ha asociado y autenticado con un AP, las tramas de control ayudan a la una correcta comunicación, algunos de los frames de control son: ACK, RTS, CTS y las tramas libres de contienda (tramas enviadas con la función PCF).
- Datos

3.3 Capa PHY del estándar IEEE 802.11

3.3.1 Arquitectura de la capa PHY

La capa física definida en el estándar IEEE 802.11 está dividida en 2 subcapas: PLCP (Physical Layer Convergence Procedure) y PMD (Physical Medium Dependent), la estructura del estándar se muestra en la Figura 3.6.

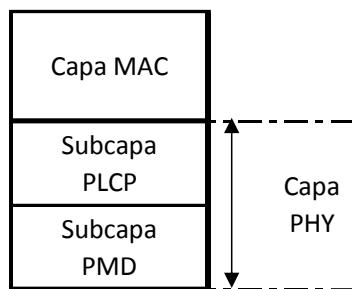


Figura 3.6 .- Estructura del estándar IEEE 802.11

La capa MAC se comunica con la subcapa PLCP, la cual define el método para el mapeo de los frames provenientes de la capa MAC llamados MPDUs (MAC layer Protocol Data Units) en un formato de trama apropiada para ser transmitida y recibida entre dos o más estaciones usando la subcapa PMD. La función de la capa PMD es definir las características y el método de transmisión y recepción de datos por el medio inalámbrico. Además de las subcapas PLCP y PMD, existe una unidad llamada PLME (Physical Layer Management Entity) la cual es la encargada de realizar funciones de gestión de capa física.

3.3.2 Esquemas de transmisión

El estándar IEEE 802.11 en su versión original publicada en 1997 incluyó 3 técnicas de transmisión para capa física, dos de ellas se trataban de técnicas de espectro disperso (FHSS y DSSS) y una más basada en radiación Infrarroja (IR).



- **DSSS (Direct Sequence Spread Spectrum):** Opera en la banda ISM de 2.4 GHz con velocidades de transmisión de 1 Mbps y 2 Mbps, soporta el uso simultáneo de hasta 3 canales no superpuestos, cada uno con un ancho de banda de 5 MHz. Opera con un esquema de modulación DBPSK para tasas de transmisión de 1 Mbps y modulación DQPSK para tasas de 2 Mbps.

La tecnología DSSS se encarga de distribuir la potencia de salida en un ancho de banda amplio, combinando el mensaje a transmitir con una secuencia pseudo-aleatoria también llamada secuencia Barker de longitud 11, es decir, cada bit de datos '1' o '0' es mapeado a su secuencia equivalente compuesta de 11 bits.

- **FHSS (Frequency Hopping Spread Spectrum):** Al igual que la tecnología DSSS, FHSS para el estándar 802.11 opera en la banda ISM de 2.4 GHz a tasas de transmisión de 1 y 2 Mbps. El número de canales disponibles es 79 con un ancho de banda de 1 MHz.

Los sistemas que trabajan con este esquema de transmisión envían ráfagas de información en distintos canales de frecuencias, por lo que la señal de transmisión salta de un canal a otro y la secuencia de canales en la cual se realiza la transmisión es una secuencia pseudo-aleatoria que distribuye la señal uniformemente en toda la banda de frecuencias usada.

- **IR (Infrared Light):** El esquema infrarrojo del estándar IEEE 802.11 es omnidireccional. Se utiliza modulación 16-PPM (Pulse Position Modulation) en donde el pulso transmitido tiene una amplitud y ancho fijos, en cambio la posición del pulso en el tiempo es variable, lo cual reduce la potencia necesaria para la comunicación. En la modulación 16-PPM cada grupo de 4 bits es mapeado a uno de los 16 símbolos PPM.

Debido a la creciente demanda de servicios de comunicaciones más eficientes, con mayores tasas de transmisión, se han definido otras versiones del estándar IEEE 802.11 las cuales cuentan con otras tecnologías de capa PHY que trabajan con distintos esquemas de transmisión, distintas frecuencias y anchos de banda, como es el caso del esquema OFDM introducido en la versión 'a' de la norma, algunas de las variantes más importantes del estándar se muestran en la Figura 3.7.

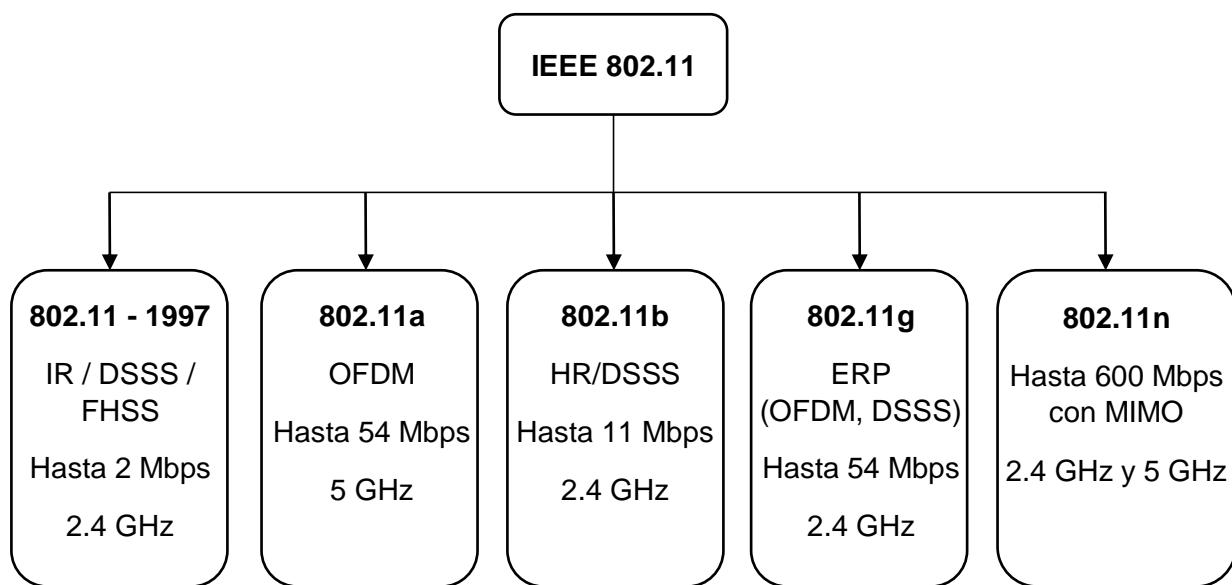


Figura 3.7 .- Versiones del estándar IEEE 802.11 para redes WLAN

3.4 El estándar IEEE 802.11g

3.4.1 Especificación ERP

En el año 2003 se publicó una variante del estándar IEEE 802.11, en la cual se alcanzaban velocidades de transmisión mayores en comparación con la versión 'b' de hasta 54 Mbps, la cual es comparable con la máxima tasa de transmisión de la versión 'a', pero a diferencia de ésta que opera en la banda de 5 GHz, la revisión 'g' opera a una frecuencia de 2.4 GHz con un ancho de banda de 20 MHz.



El esquema de capa física introducido por la versión 'g' adoptó el nombre de ERP (Extended Rate PHY) pues presenta una mayor variedad de velocidades de datos y esquemas de modulación. Las tecnologías de capa física definidas bajo la especificación ERP son:

- **ERP-DSSS/CCK**: Se refiere a la capa física usada en el estándar IEEE 802.11b. Se usa la tecnología DSSS con la modulación CCK. Soporta velocidades de 1, 2, 5.5 y 11 Mbps.
- **ERP-OFDM**: Éste esquema usa modulación OFDM para ofrecer las mismas velocidades de transmisión del estándar IEEE 802.11a (6, 9, 12, 18, 24, 36, 48, 54 Mbps) en la banda de frecuencias comprendida entre 2.4 y 2.4835 GHz.
- **ERP-PBCC**: Esta capa física se introdujo en la versión IEEE 802.11b, usa tecnología DSSS con codificación PBCC (Packet Binary Convolutional Code). Soporta velocidades de 5.5, 11, 22 y 33 Mbps.
- **DSSS-OFDM**: Se trata de una tecnología de capa física híbrida, pues combina los esquemas DSSS y OFDM. El encabezado se envía con la tecnología DSSS, mientras que los datos útiles se transmiten bajo el esquema OFDM. Ofrece las mismas tasas de transmisión que la especificación ERP-OFDM.

3.4.2 OFDM

La multiplexión por división de frecuencias ortogonales (OFDM), como su nombre lo indica, es un tipo de sistema multiportadora en donde los diferentes canales de datos comparten el ancho de banda disponible.

En OFDM la señal está compuesta de distintos canales independientes (subportadoras) igualmente espaciados, cada uno de los cuales usa una fracción del ancho de banda para transportar información.

Una de las principales razones para el uso de OFDM es debido a que los sistemas OFDM son robustos frente a canales selectivos en frecuencia [19].

En sistemas con una sola portadora, la presencia de cualquier interferencia provoca que el sistema de comunicación falle por completo, en cambio, en sistemas multiportadora, sólo un pequeño porcentaje de portadoras son afectadas y los datos que se encuentren en error, provenientes de las subportadoras dañadas, pueden ser recuperados por medio de códigos correctores de errores.

Cada una de las subportadoras que conforman el sistema OFDM son ortogonales entre sí, impidiendo así interferencia entre portadoras adyacentes, a la vez que permite un uso eficiente del ancho de banda [20], en contraste con los sistemas FDM (Frequency Division Multiplexing) en donde se requiere el uso de bandas de guarda, como se muestra en la Figura 3.8.

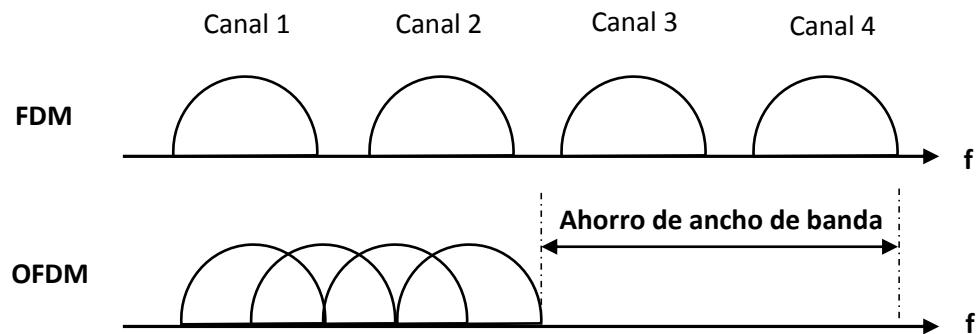


Figura 3.8 .- Comparación del uso de ancho de banda para sistemas FDM y OFDM

Uno de los problemas más comunes en sistemas de comunicación inalámbricos es la interferencia entre símbolos o ISI (Inter-Symbol Interference), provocada por el llamado efecto de propagación por multirayectorias, el cual provoca que múltiples copias de la señal transmitida lleguen al receptor en diferentes instantes de tiempo, causando que algunas de estas señales interfieran con los siguientes símbolos transmitidos.

Con el objetivo de contrarrestar la interferencia entre símbolos en los sistemas OFDM se agrega un tiempo de guarda, también llamado prefijo cíclico CP (Cyclic Prefix), en donde cada símbolo OFDM es precedido por una extensión periódica igual a 1/4 de la misma señal (Figura 3.9), lo que evita que símbolos OFDM adyacentes se interfieran entre sí.

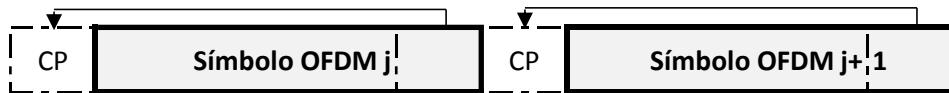


Figura 3.9 .- Inserción del ciclo prefijo en sistemas OFDM

En resumen las ventajas de un sistema OFDM y por las cuales se seleccionó como la base para el desarrollo del presente trabajo de tesis, son [19]:

- Eficiente para lidiar con multirayectorias
- Robusto contra interferencias de banda angosta
- Alta eficiencia espectral

Un modulador OFDM se implementa dividiendo un flujo de datos digitales s en otras N líneas de datos (s_k) de menor velocidad por medio de un convertidor serial-paralelo.

Cada línea de datos resultante del convertidor S/P corresponde a la información que será transmitida en una subportadora del sistema OFDM.

Una vez dividida la información por canales del sistema OFDM se toman los datos digitales para cada subportadora y se modulan (por ejemplo con modulación en fase, en amplitud o ambas) lo que da como resultado un número complejo.

Para crear el símbolo OFDM se usa la transformada rápida de Fourier inversa o IFFT (Inverse Fast Fourier Transform), el símbolo OFDM es una señal compuesta de la información de todas las subportadoras.

Finalmente se inserta el prefijo cíclico al inicio de cada símbolo OFDM. El diagrama a bloques de un transmisor OFDM básico se muestra en la figura 3.10.

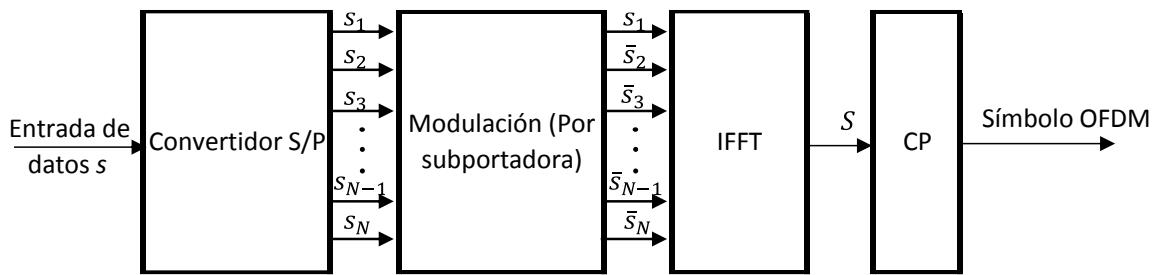


Figura 3.10 .- Diagrama a bloques de un transmisor OFDM

Para el caso del receptor OFDM se debe aplicar el proceso inverso al del transmisor. En primer lugar se debe extraer el prefijo cíclico, posteriormente se tiene que aplicar la transformada FFT (Fast Fourier Transform) para así poder extraer la información de cada una de las subportadoras y demodular la información contenida en cada canal.

Finalmente se une cada línea de información en un solo flujo de datos por medio de un convertidor paralelo-serie.

El diagrama a bloques de un receptor OFDM se muestra en la Figura 3.11.

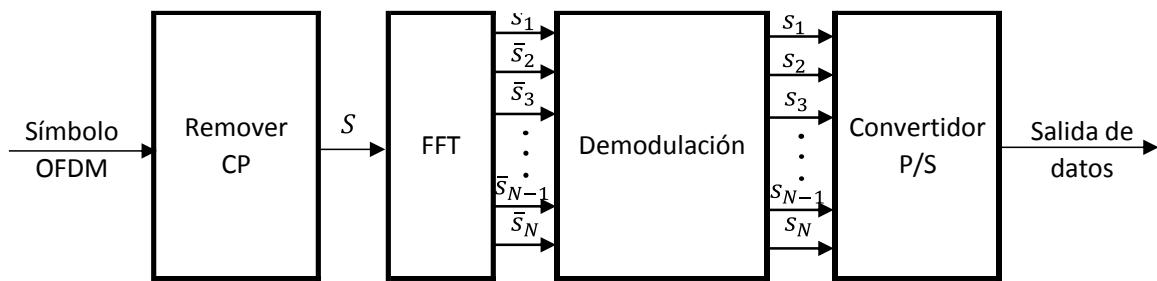


Figura 3.11 .- Diagrama a bloques de un receptor OFDM

3.4.3 Descripción de la capa PHY para el esquema OFDM

Como se definió anteriormente, la capa PLCP es la encargada de definir el método para el mapeo de los frames provenientes de la capa MAC a un formato de trama apropiado para ser transmitido por el medio inalámbrico. La Figura 3.12 presenta el formato de un PPDU (PLCP Protocol Data Unit), el cual está formado por el preámbulo PLCP, un encabezado, bits de cola y de relleno.

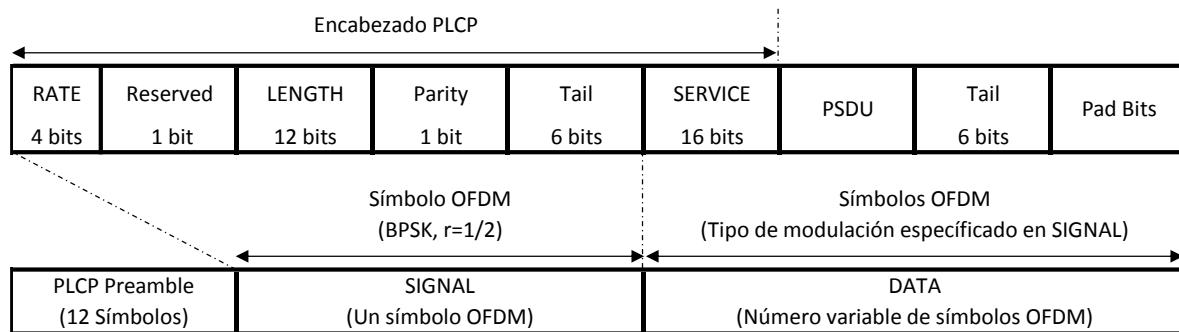


Figura 3.12 .- Formato de la trama PPDU

PLCP Preamble: El preámbulo PLCP es usado para la sincronización. Está formado de dos partes, la primera parte consiste de 10 símbolos cortos conocidos como secuencias de entrenamiento cortas, la segunda parte consiste de dos símbolos conocidos como secuencias de entrenamiento largas. La Figura 3.13 muestra la estructura del preámbulo PLCP, como se observa, el preámbulo tiene una duración de 16us.

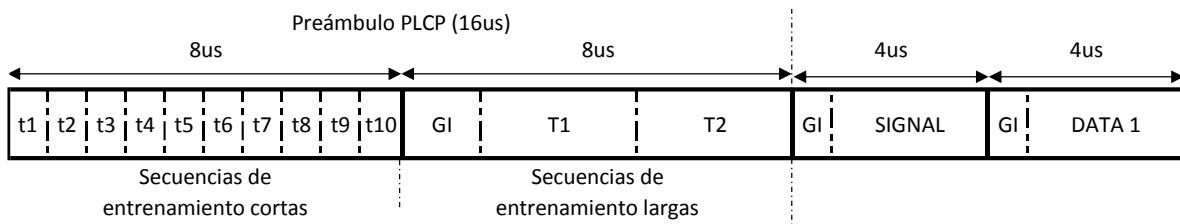


Figura 3.13 .- Secuencias de entrenamiento que conforman el preámbulo PLCP

Las primeras 7 secuencias de entrenamiento cortas ($t_1 - t_7$) son destinadas a la detección de la señal en el receptor y al control automático de ganancia, mientras que las últimas 3 son usadas para una corrección en frecuencia burda.

Las secuencias de entrenamiento largas tienen el objetivo de servir para la estimación del canal y para una fina corrección en frecuencia.

La secuencia de entrenamiento corta consiste de 12 subportadoras de las 52 subportadoras de datos disponibles (hay 64 subportadoras en total, 12 son nulas y 52 son de datos). De tal forma que una secuencia de entrenamiento corta normalizada en el dominio de la frecuencia está dada por:

$$S_{-26,26} = \sqrt{\frac{13}{6}} \{0,0,1+j,0,0,0,-1-j,0,0,0,1+j,0,0,0,-1-j,0,0,0,-1-j,0,0,0,1+j,0,0,0,0,0,0,-1-j,0,0,0,1+j,0,0,0,1+j,0,0,0,1+j,0,0,0,1+j,0,0\}$$

La secuencia de entrenamiento larga, se puede definir en frecuencia como:

$$S_{-26,26} = \{ 1,1,-1,-1,1,1,-1,1,-1,1,1,1,1,1,1,1,-1,-1,1,1,-1,1,-1,1,1,1,1,1,0,1, \\ -1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,1,1,-1,-1,1,-1,1,-1,1,-1,1,-1,1,1,1,1,1\}$$



Finalmente cada secuencia de entrenamiento es transformada por la IFFT, a las secuencias de entrenamiento largas se les agrega el prefijo cíclico.

SIGNAL: El campo de señal usa un símbolo OFDM, el cual contiene información relevante para que el receptor pueda decodificar apropiadamente los datos enviados por el transmisor, tales como la tasa de codificación y la longitud de los datos enviados.

Este campo es modulado con modulación BPSK y una tasa de codificador convolucional de 1/2, por lo que se debe aplicar un proceso similar al del campo de datos, excepto por el randomizador.

El campo de señal debe codificarse usando el codificador convolucional 1/2, pasar por el interleaver, modularse (usando BPSK) y se le deben insertar las portadoras piloto y finalmente aplicarse la transformada inversa de Fourier, dichos procesos se describen a detalle en las sección posterior.

La estructura del campo de señal se muestra en la Figura 3.14.

B0	B3	B4	B5	B16	B17	B18	B23
RATE 4 bits	RES. 1 bit		LENGTH 12 bits	PARITY 1 bit	SIGNAL TAIL 6 bits		

Figura 3.14 .- Estructura del campo SIGNAL con longitud de 24 bits

Como se aprecia en la figura 3.14, la parte de la señal está compuesta de:

- RATE: Indica la velocidad a la cual se están transmitiendo los datos, de acuerdo a la Tabla 3.1. Como se puede observar, al indicar la tasa de transmisión, implícitamente se define el tipo de modulación y la tasa del codificador convolucional usado.



Tasa (Mbps, BW= 20 MHz)	Bits (B0 - B3)	Esquema de Modulación
6	1101	BPSK 1/2
9	1111	BPSK 3/4
12	0101	QPSK 1/2
18	0111	QPSK 3/4
24	1001	16-QAM 1/2
36	1011	16-QAM 3/4
48	0001	64-QAM 2/3
54	0011	64-QAM 3/4

Tabla 3.1 .- Contenido del campo RATE dependiendo de la tasa de transmisión

- RESERVED, SIGNAL TAIL: Estos campos contienen un valor de '0'.
- LENGTH: Indica la longitud en bytes de la trama PSDU, contiene 12 bits.

DATA: Cada símbolo OFDM de datos obedece la estructura que se muestra en la Figura 3.15.

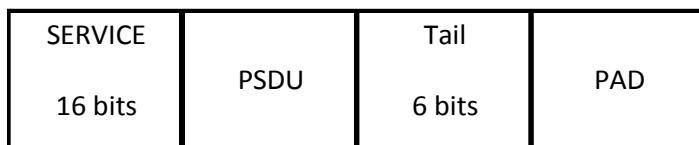


Figura 3.15 .- Estructura del campo de datos

- SERVICE: El campo de servicio de la trama de datos contiene 16 bits en '0', los primeros 7 bits son usados para inicializar el randomizador en el receptor y los últimos 10 bits están reservados para uso futuro.
- PSDU: Se trata de la trama proveniente de la capa MAC que se desea transmitir.



- Tail: Este campo contiene 6 bits en '0' y tiene como propósito reiniciar el codificador convolucional a '0'.
- PAD: Éste campo tiene la función de agregar bits de relleno, de tal forma que el número de bits de datos a enviar sea un múltiplo entero del número de bits en un símbolo OFDM, la cantidad de bits de relleno depende del número de bits de datos y del esquema de modulación. El cálculo para los bits de relleno se realiza calculando y sustituyendo las ecuaciones (3-1) y (3-2) en la ecuación (3-3).

$$(3-1) \quad N_{SYM} = \left\lceil \frac{16 + 8 * LENGTH + 6}{N_{DBPS}} \right\rceil$$

$$(3-2) \quad N_{DATA} = N_{SYM} \times N_{DBPS}$$

$$(3-3) \quad N_{PAD} = N_{DATA} - (16 + 8 * LENGTH + 6)$$

En donde:

N_{SYM} representa el número total de símbolos OFDM de datos a transmitir.

N_{DBPS} es el número de bits en un símbolo OFDM y depende del esquema de modulación usado, sus posibles valores se encuentran en la tabla 3.2.

N_{DATA} se refiere al número de bits de datos (Longitud en bits del campo DATA).

N_{PAD} indica el número de bits de relleno.

Tasa (Mbps, BW=20 MHz)	Esquema de Modulación	Bits codificados por subportadora (NBPS)	Bits codificados por símbolo OFDM (NCBPS)	Bits de Datos por símbolo OFDM (NDBPS)
6	BPSK 1/2	1	48	24
9	BPSK 3/4	1	48	36
12	QPSK 1/2	2	96	48
18	QPSK 3/4	2	96	72
24	16-QAM 1/2	4	192	96
36	16-QAM 3/4	4	192	144
38	64-QAM 2/3	6	288	192
54	64-QAM 3/4	6	288	216

Tabla 3.2 .- Parámetros dependientes de la tasa de transmisión

Una vez completado el campo de datos, los datos digitales deben ser procesados por una serie de bloques antes de poder ser enviados por el medio inalámbrico, éstos son:

- Scrambler: En primer lugar, los datos deben mezclarse por medio de un randomizador, para evitar la presencia de '1' lógicos consecutivos. El polinomio generador es $S(x) = x^7 + x^4 + 1$. El diagrama del scrambler se muestra en la Figura 3.16.

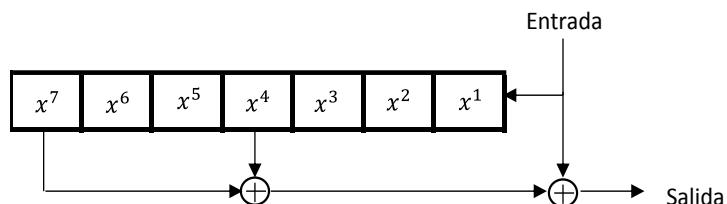


Figura 3.16 .- Scrambler de datos

- Codificador Convolucional: El codificador convolucional es un código corrector de errores, el cual añade bits de redundancia al mensaje, lo que permite recuperar el mensaje original en el receptor si algunos bits se dañan durante la transmisión.

Para lograr una tasa de codificación específica, el estándar IEEE 802.11g define la codificación en 2 partes, por un lado un codificador convolucional con tasa 1/2 (con polinomios generadores 133₈ y 171₈) y por otra parte un bloque de acortamiento o *puncturing* para lograr las tasas de 2/3 y 3/4.

El diagrama del codificador convolucional con tasa 1/2 se muestra en la Figura 3.17, en donde como se puede observar, por cada bit de entrada X_n hay dos bits de salida A_n y B_n .

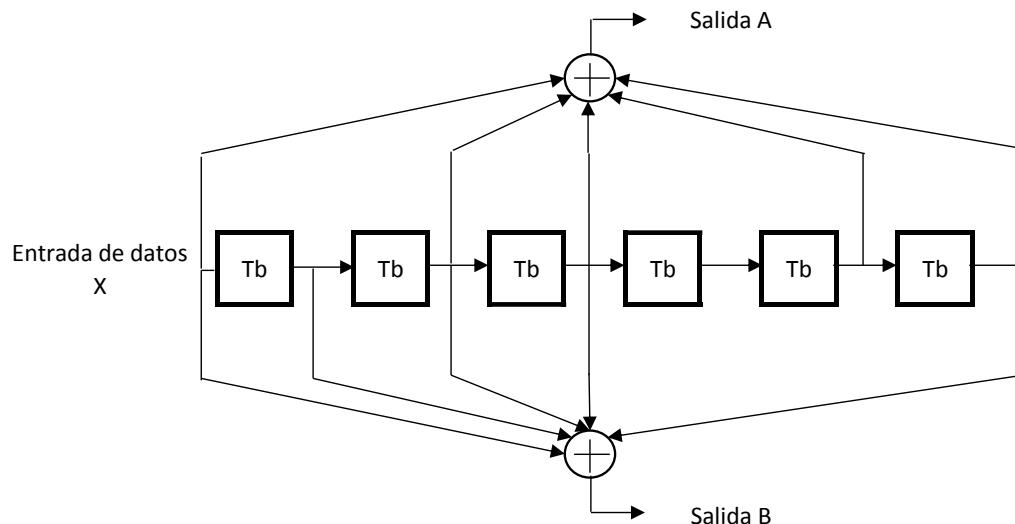


Figura 3.17 .- Diagrama del Codificador convolucional, r=1/2

Para el caso de modulaciones con tasa 1/2, todos los bits procedentes del codificador son transmitidos, en cambio para alcanzar las tasas 2/3 y 3/4 se requiere realizar un proceso de acortamiento, en donde los bits de ciertas posiciones no son transmitidos. El proceso de puncturing para tasa $r=3/4$ se muestra en la Figura 3.17 y para $r=2/3$ en la Figura 3.18.

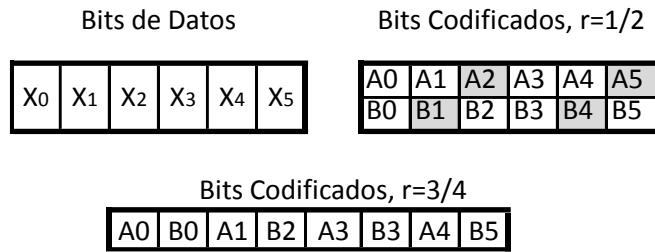


Figura 3.17 .- Proceso de Puncturing para $r=3/4$

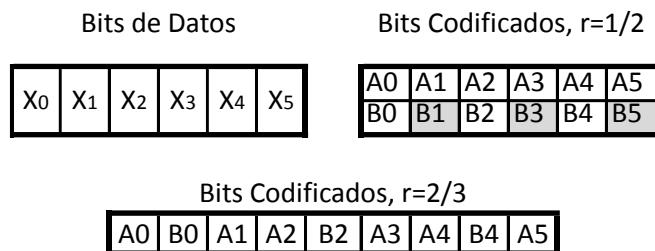


Figura 3.18 .- Proceso de Puncturing para $r=2/3$

- Interleaver: Los datos deben pasar por un bloque entrelazador, el cual permite aislar los errores de ráfaga permitiendo que puedan ser corregidos mediante el codificador convolucional.

La función de entrelazado del estándar IEEE 802.11 está formada por dos permutaciones. La primera permutación permite que bits adyacentes sean transmitidos en subportadoras no adyacentes. La segunda permutación alterna los bits en posiciones más y menos significativas de una constelación.

Ambas permutaciones están en función de la posición original y del número de bits codificados por símbolo OFDM (N_{CBPS}) que a su vez depende del tipo de modulación, como se puede observar en la Tabla 3.2.

La nueva posición de cada bit, después de la primera permutación, está representada por i en la ecuación (3-4); mientras que la posición, después de la segunda permutación, está representada por j de la ecuación (3-5).

$$(3-4) \quad i = \left(\frac{N_{CBPS}}{16} \right) (k \bmod 16) + \lfloor k/16 \rfloor \quad k = 0, 1, \dots, N_{CBPS} - 1$$

$$(3-5) \quad j = s * \left\lfloor \frac{i}{s} \right\rfloor + \left(i + N_{CBPS} - \left\lfloor 16 * \frac{i}{N_{CBPS}} \right\rfloor \right) \bmod s \quad i = 0, 1, \dots, N_{CBPS} - 1$$

Donde: $s = \max(N_{BPSC}/2, 1)$

La cadena que contiene el mensaje de salida del entrelazador debe dividirse en bloques de N_{CBPS} bits cada uno, cada bloque de N_{CBPS} bits representa la información que será enviada en un símbolo OFDM.

- Modulador: Una vez formados los grupos de N_{CBPS} bits, las subportadoras OFDM se deben modular usando BPSK, QPSK, 16-QAM o 64-QAM según sea el caso, la Figura 3.19 muestra las constelaciones usadas para la modulación BPSK y 16-QAM.

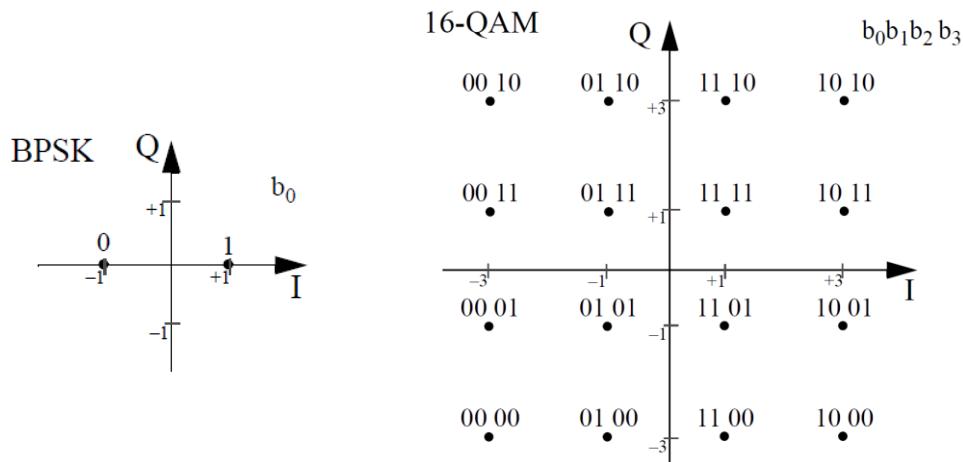


Figura 3.19 .- Constelaciones para modulación BPSK y 16-QAM



Una vez que se han modulado los datos, los valores complejos obtenidos deberán ser normalizados, de acuerdo a los valores de la tabla 3.3 definida por el estándar.

Modulación	Factor
BPSK	1
QPSK	$1/\sqrt{2}$
16-QAM	$1/\sqrt{10}$
64-QAM	$1/\sqrt{42}$

Tabla 3.3 .- Factor de normalización después de la modulación

- Inserción de portadoras piloto: Hasta este momento cada símbolo OFDM debe estar formado por 48 números complejos, se deben agregar 4 portadoras piloto en las posiciones -21, -7, 7, 21 (con valores 1, 1, 1, -1 respectivamente) alternando su polaridad. Adicionalmente se deben agregar la portadora de DC y las subportadoras en los extremos (-32 a -27 y +27 a +31) con un valor de 0.
- IFFT: Se realiza la transformada inversa de Fourier de las 64 subportadoras para generar el símbolo OFDM en el dominio del tiempo.
- CP: Se agrega el ciclo prefijo a cada símbolo OFDM anteponiendo la última cuarta parte de cada símbolo para contrarrestar los efectos de las multirayectorias.



CAPÍTULO IV. SDR

Se puede definir a un radio definido por software (SDR) como el conjunto de componentes en software y hardware que permiten la implementación de un sistema de comunicaciones inalámbrico con parámetros reconfigurables vía software, tales como la frecuencia de operación, el ancho de banda, el tipo de modulación, codificación y la potencia de salida.

Este capítulo se encuentra dividido en dos secciones, el objetivo de la primera parte es presentar una introducción a la plataforma SDR, su definición y abordar sus características principales.

La segunda sección pretende brindar una descripción de la plataforma SDR usada para la implementación de éste proyecto, el equipo USRP N210, se presenta su arquitectura, así como sus características más importantes.

4.1 Plataforma SDR

4.1.1 Introducción

El término *Software Radio* (SR) fue introducido por Joseph Mitola en 1991, en donde describió a un SR como un dispositivo en donde los convertidores D/A (Digital - Analógico), A/D (Analógico – Digital), las antenas transmisora y receptora y el dispositivo en general permiten que el radio lleve a cabo las funciones de transmisión, recepción, generación de señal, modulación/demodulación, sincronización, control y codificación/decodificación vía software [9].

El radio definido por software ideal, según J. Mitola, debe ser capaz de interoperar con cualquier servicio de comunicaciones. Seleccionando un algoritmo específico, el radio instantáneamente debe reconfigurarse para soportar el tipo de servicios deseados [9].



Uno de los principales problemas que enfrenta la industria de las comunicaciones inalámbricas es debido a la constante evolución de las tecnologías y protocolos que corresponden a las capas más bajas del modelo OSI (capa PHY y capa LLC) [21].

Los equipos de radio tradicionales deben reemplazarse cuando se requiere actualizar el sistema a una nueva generación de estándares o simplemente migrar a otro tipo de tecnología, por lo que se deben buscar alternativas viables, entre las que destaca la implementación de un sistema basado en SDR por ser una solución eficiente y económica en el desarrollo de sistemas de comunicaciones inalámbricos multimodo.

Entre las ventajas que la tecnología SDR presenta [22], se destacan:

- Es reconfigurable: La plataforma SDR permite que distintos estándares coexistan en el mismo sistema de comunicación, en diferentes módulos de software, permitiendo una configuración dinámica del sistema con sólo seleccionar el módulo de software apropiado.
- Es interoperable: La tecnología SDR facilita la implementación de sistemas de radio con arquitectura abierta, con lo cual un mismo dispositivo puede comunicarse con dispositivos con otras tecnologías simultáneamente.
- Es confiable: La confiabilidad del sistema SDR depende del software con el que opere, puesto que se pueden establecer algoritmos para la prevención, detección y corrección de errores.

Sin embargo, al implementar un sistema de comunicaciones con SDR, se debe tomar en cuenta que a comparación de los radios tradicionales, un radio definido por software consume una mayor potencia debido a la mayor cantidad de procesamiento que realiza, además, implica un mayor costo inicial, por lo que se debe analizar la aplicación en donde se implementará.

4.1.2 Diagrama a bloques

El diagrama a bloques general de un sistema SDR [21] se muestra en la Figura 4.1.

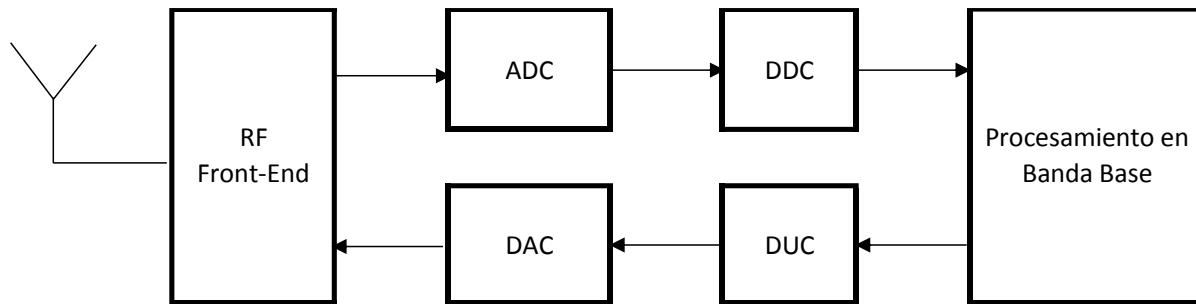


Figura 4.1.- Diagrama a bloques de un sistema SDR

Las funciones que realiza cada uno de los bloques presentados en la Figura 4.1 son:

- El bloque *RF Front-End* esencialmente está formado de componentes analógicos, como su nombre lo indica, es el encargado de enviar y recibir las señales de radiofrecuencia (RF) a través de la antena.

En general, éste bloque realiza el procesamiento de la señal de RF, está formado por filtros, amplificadores de bajo ruido (recepción), amplificadores de potencia (transmisión) y osciladores locales, además, realiza la conversión entre radio frecuencia y frecuencia intermedia (IF).

La frecuencia intermedia es una frecuencia más baja que la radio frecuencia para que los convertidores A/D y D/A puedan realizar la digitalización y viceversa.



- Los bloques *ADC* (Analog to Digital Converter) en el receptor y *DAC* (Digital to Analog Converter) en el transmisor, como su nombre lo indica, realizan la interface entre la parte analógica y la parte digital del sistema.
- El bloque *DDC* (Digital Down Converter) en el receptor convierte la señal muestreada (centrada en una frecuencia intermedia) a una señal en banda base, en cambio el bloque *DUC* (Digital Down Converter) en el transmisor, realiza el proceso inverso.
- Finalmente el bloque de procesamiento en banda base se encarga de la modulación, ecualización de la señal recibida, sincronización, correlación, etc. El procesamiento en banda base puede realizarse por DSPs o FPGAs.

4.1.3 Clasificación

De acuerdo a su arquitectura, los sistemas SDR se pueden clasificar en [23]:

- Enfoque de una computadora de propósito general: En este tipo de plataforma SDR, se usa una PC (Personal Computer) de propósito general, la cual permite un proceso flexible y de baja complejidad de programación y configuración del sistema, pero con un alto consumo de energía. Es el tipo de arquitectura más usado para aplicaciones de investigación en el que los tiempos de desarrollo y reconfiguración son cortos.

Entre las plataformas SDR comerciales con éste tipo de arquitectura destacan:

USRP [24]: El *Universal Software Radio Peripheral*, se compone de ADC/DAC de alta frecuencia que muestran a frecuencia intermedia, un FPGA que realiza el procesamiento y almacenamiento de la señal en banda base.



Para el procesamiento de la señal en banda base, el USRP se conecta con una PC mediante USB (Universal Serial Bus) para el caso del USRP 1 o por medio de la tarjeta de red con el USRP N210.

Quicksilver [25]: La plataforma Quicksilver es en estructura similar al USRP, sin embargo es solamente capaz de recibir señales RF.

Microsoft SORA [26]: Se conecta a una PC mediante la interfaz PCIe (Peripheral Component Interconnect Express).

- Enfoque de coprocesador: Con el objetivo de acelerar el procesamiento de la señal, a las computadoras de propósito general se agrega un co-procesador (FPGA o DSP).
El co-procesador se encarga de la ejecución de grandes cantidades de procesamiento. De esta forma se reduce el consumo de energía respecto al enfoque de la computadora de propósito general, aunque su flexibilidad se ve un poco afectada, puesto que el lenguaje de programación puede variar entre fabricantes.
- Enfoque de procesador central: Con el objetivo de obtener plataformas eficientes se usan procesadores dedicados (DSP o FPGA) para realizar el procesamiento de la señal en banda base. Generalmente se usa uno o varios procesadores para el procesamiento de la señal y además un procesador central para el control del sistema. La desventaja de este tipo de arquitectura es que es muy poco flexible.

Debido a que en éste proyecto no se tiene ninguna limitación en cuanto al consumo de potencia, se optó por una plataforma comercial que trabajara con una computadora de propósito general, como es el caso del equipo USRP N210.

4.2 USRP N210

4.2.1 Arquitectura

El *Universal Software Radio Peripheral* o USRP N210 es un tipo de SDR desarrollado por la compañía Ettus Research, perteneciente a National Instruments, diseñado con el objetivo de ser una plataforma flexible y de bajo costo, que opera junto a una computadora de propósito general. El USRP N210 se muestra en la Figura 4.2.



Figura 4.2.- Imagen del equipo USRP N210

El diagrama del USRP N210 se muestra en la Figura 4.3 [27].

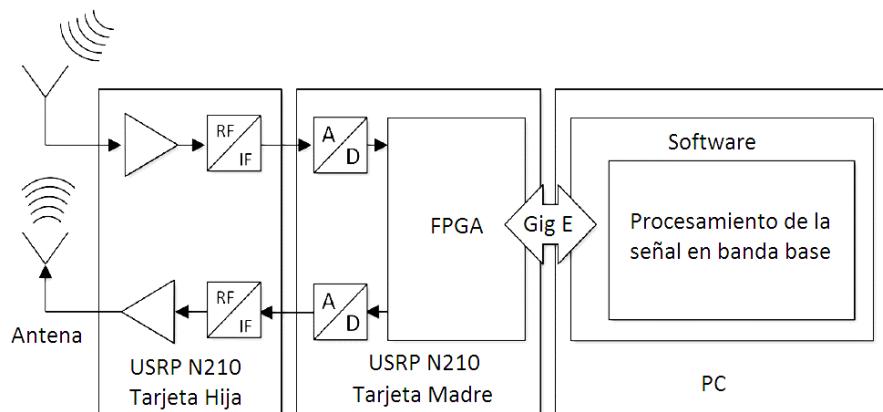


Figura 4.3.- Diagrama a bloques del USRP N210

Como se puede observar, el USRP está formado por un lado, por la etapa de procesamiento de la señal en radiofrecuencia, la cual se realiza en la **tarjeta hija**.

Por otro lado, la **tarjeta madre** contiene los conversores DAC y ADC así como un FPGA que permite la comunicación con una computadora para el procesamiento de la señal en banda base. La comunicación con la PC se realiza mediante una interfaz **Gigabit Ethernet**, a diferencia de su antecesor el USRP 1, el cual operaba mediante una interfaz USB.

El panel frontal del USRP N210 se muestra en la Figura 4.4

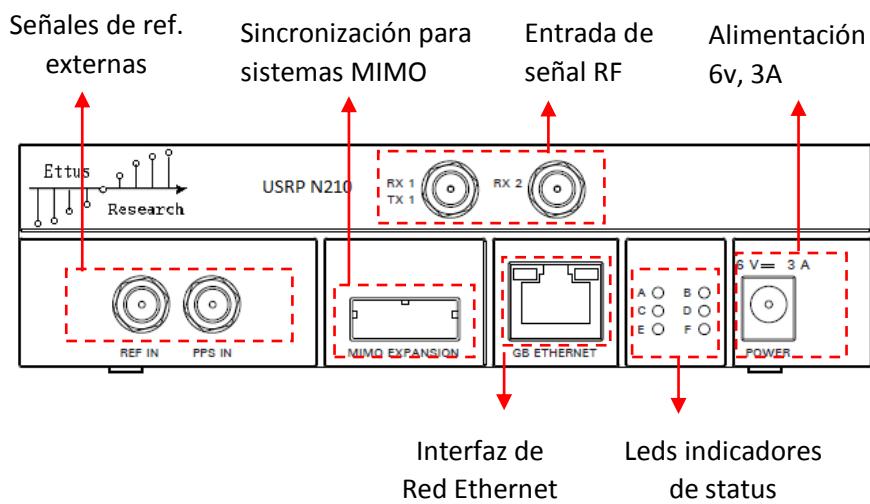


Figura 4.4.- Panel frontal del USRP N210

Una vez que se tienen listas las muestras en banda base, éstas son enviadas al driver del USRP, el cual es llamado **UHD** (USRP Hardware Driver), el cual toma las muestras y las inserta en tramas Ethernet para ser enviados a la PC y es compatible con sistemas operativos Linux, MacOS y Windows.

Como se mencionó anteriormente la **PC** es la encargada del procesamiento en banda base, para tal tarea se pueden usar tres distintas plataformas de programación con el USRP:



- LabVIEW: Es un software desarrollado por National Instruments, el cual usa lenguaje de programación gráfico. LabVIEW permite la implementación de sistemas de automatización, control y procesamiento de datos permitiendo usar distintas plataformas de hardware en los sistemas desarrollados.
- GNU Radio: Es un software de desarrollo de código abierto que provee bloques de procesamiento de señales para ser implementados con plataformas de hardware o en ambientes de simulación. Los bloques de GNU Radio son programados en lenguaje C++, se conectan unos con otros mediante lenguaje Python, además la apariencia de cada bloque debe especificarse en lenguaje XML.
- Matlab: Es un lenguaje de programación de alto nivel desarrollado por MathWorks que de igual manera que las dos plataformas anteriores, brinda bloques para el procesamiento de señales y contiene bloques para trabajar con la plataforma SDR.

En el presente proyecto se eligió el lenguaje de programación LabVIEW para la implementación de la parte de procesamiento de la señal en banda base proveniente del USRP.

Una de las razones por las que se eligió el software LabVIEW fue debido a que en la revisión de los desarrollos hechos bajo la plataforma SDR se pudo observar que actualmente no hay estudios a fondo sobre la implementación de un sistema robusto de comunicaciones con éste lenguaje de programación.

Adicionalmente, debido a la naturaleza gráfica de LabVIEW, en comparación con el software GNU Radio, el desarrollo y programación de bloques individuales de procesamiento digital se lleva a cabo de una forma más intuitiva, por lo que el tiempo de desarrollo, implementación y posterior modificación es menor.

4.2.2 Tarjeta madre

La tarjeta madre es la parte medular del USRP, en ella se conectan todas las interfaces del USRP (el puerto de expansión MIMO, las señales de referencia externas, la interfaz Ethernet) y la tarjeta hija. El componente principal de la tarjeta madre es el FPGA cuya principal función es la conversión entre IF y banda base. La tarjeta madre también contiene 2 conversores DAC y ADC y sus respectivas etapas DDC y DUC.

La Figura 4.5 muestra la tarjeta madre del USRP N210.



Figura 4.5.- Tarjeta madre del USRP N210

Las características principales de la tarjeta madre del USRP N210 se muestran a continuación en la Tabla 4.1 [28].

Parámetro	Valor
ADC Frecuencia de Muestreo	100 MSPS
ADC Resolución	14 bits
DAC Frecuencia de Muestreo	400 MSPS
DAC Resolución	16 bits
Host Frecuencia de Muestreo	50 MSPS
FPGA	Spartan 3400
Interface	Gigabit Ethernet
Alimentación	6v, 3A

Tabla 4.1.- Características de la tarjeta madre

4.2.3 Tarjeta hija SBX

Como se mencionó anteriormente, la tarjeta hija en el USRP realiza las funciones del procesamiento en radiofrecuencia.

La plataforma USRP N210 brinda una gran variedad de tarjetas hijas, las cuales tienen diferentes características en cuanto a bandas de frecuencia, ganancias, si actúan como transmisor, receptor o ambas.

La tarjeta hija que se usó para el desarrollo del receptor OFDM propuesto es éste trabajo de tesis es la tarjeta SBX, la cual trabaja en un rango de frecuencias de entre los 400 MHz y los 4400 MHz, con ganancias configurables desde 0 hasta 31.5 dB tanto para la recepción como para la transmisión. El máximo ancho de banda que soporta es de 40 MHz.

La Figura 4.6 muestra la tarjeta hija SBX.



Figura 4.6.- Tarjeta hija SBX



CAPÍTULO V. SOFTWARE LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un lenguaje de programación gráfico desarrollado por National Instruments que usa iconos en lugar de líneas de texto para crear aplicaciones. El software LabVIEW proporciona extensas herramientas necesarias para construir cualquier aplicación de medición o control. NI LabVIEW es un entorno de desarrollo ideal para la innovación y descubrimiento, pues es posible combinar la potencia del software LabVIEW con hardware modular y reconfigurable.

El presente capítulo presenta una descripción del software LabVIEW, el cual fue seleccionado para realizar la programación de la capa física del estándar IEEE 802.11 con el USRP N210. En éste capítulo se definen las características principales de LabVIEW, se brinda una descripción del lenguaje de programación, las herramientas más comunes y finalmente en la última sección se describen los bloques que permiten la configuración del USRP.

5.1 El entorno de programación de LabVIEW

5.1.1 Introducción

En contraste con los lenguajes de programación basados en texto, donde las instrucciones determinan el orden de la ejecución del programa, LabVIEW usa programación por flujo de datos.



El flujo de datos a través de los nodos de un diagrama a bloques determina el orden de ejecución de las funciones y VIs (Virtual Instruments). [29]

La naturaleza gráfica de LabVIEW lo hace ideal para ser usado en el desarrollo de aplicaciones de automatización, control de instrumentos, adquisición y procesamiento de datos, etc.

5.1.2 Instrumentos Virtuales

En general el término Instrumento Virtual (VI) se refiere a un elemento programado en LabVIEW que se encarga de tomar datos de entrada, ya sea desde otros nodos o de la interfaz del usuario, procesarlos y enviarlos a otros bloques o desplegarlos en pantalla.

Un VI consta de 3 elementos fundamentales:

- Panel frontal
- Diagrama a bloques
- Icono y panel de conexión

Panel Frontal

El panel frontal es la interfaz del usuario, en él se encuentran los *indicadores* los cuales permiten desplegar las salidas o datos obtenidos en pantalla y son usados para monitorear el instrumento virtual y los *controles*, los cuales permiten la entrada de datos interactiva por parte del usuario cuando el VI se encuentra en operación (Figura 5.1).

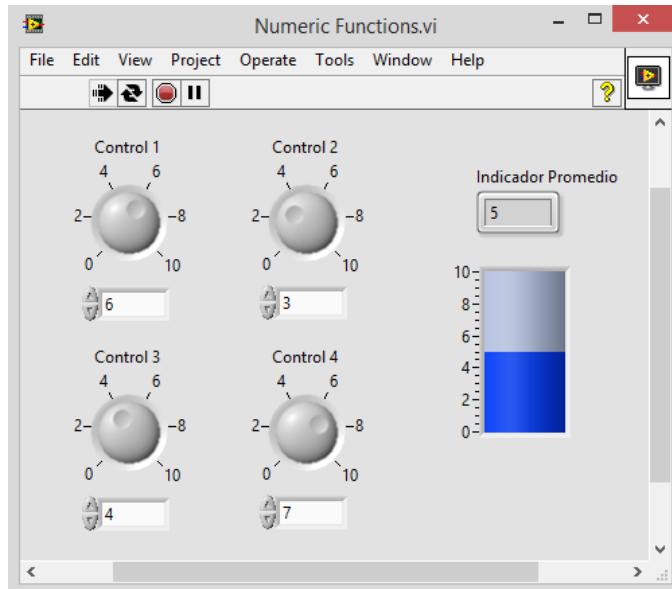


Figura 5.1 .- Panel Frontal de un VI en LabVIEW

Para construir el panel frontal de un instrumento virtual, los elementos deben ser seleccionados de la paleta de controles (Figura 5.2), la cual contiene distintos indicadores y controles, entre los que destacan:

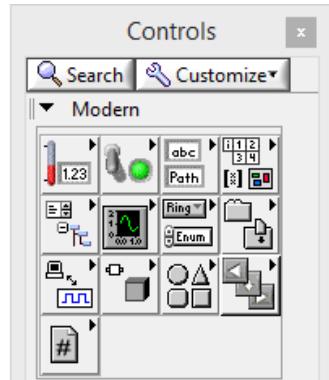


Figura 5.2 .- Paleta de Controles

- Indicadores numéricicos: Display, termómetro, tanque, medidor, barra de progreso, calibrador, etc. (Figura 5.3).

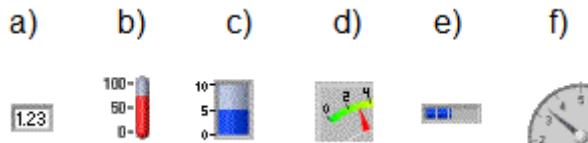


Figura 5.3.- Indicadores numéricicos. **a)** Display, **b)** Termómetro, **c)** Tanque, **d)** Medidor, **e)** Barra de progreso, **f)** Calibrador

- Controles numéricos: Estándar, perilla, selector, barra de desplazamiento (Figura 5.4).

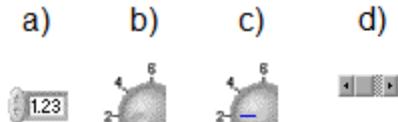


Figura 5.4.- Controles numéricos. **a)** Estándar, **b)** Perilla, **c)** Selector, **d)** Barra de desplazamiento

- Indicador Booleanos: Led circular y Cuadrado.
- Controles Booleanos: Botón de presión, interruptor basculante, interruptor de palanca, botón de stop, botón de selección, etc. (Figura 5.5).

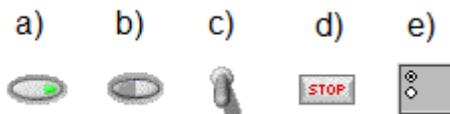


Figura 5.5.- Controles booleanos. **a)** Botón de presión, **b)** Interruptor basculante, **c)** Interruptor de palanca, **d)** Botón de stop, **e)** Botón de selección

- Control e Indicador de cadena de caracteres.
- Gráficas.



Diagrama a bloques

El diagrama a bloques es la parte modular de un VI, ya que contiene el código de programación, por lo que define las funciones realizadas por el instrumento virtual (Figura 5.6).

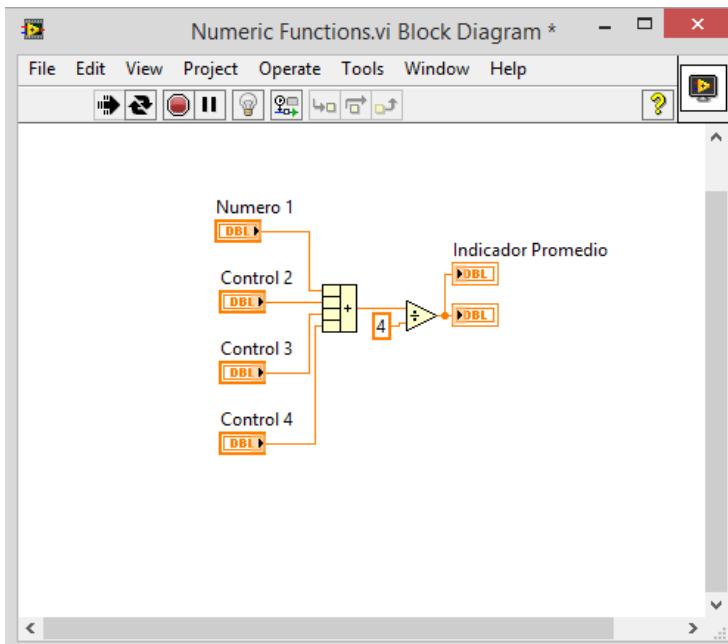


Figura 5.6 .- Diagrama a bloques de un VI en LabVIEW

Cada bloque requiere de una o más entradas para procesarla(s) y generar una salida. Al igual que en otros lenguajes de programación, en LabVIEW existen diferentes tipos de datos, los tipos de datos más importantes se muestran en la Tabla 5.1.

Los tipos de datos pueden ser distinguidos por color y apariencia, además cada uno es usado para aplicaciones específicas.



TIPO DE DATO	REPRESENTACIÓN	DESCRIPCIÓN
Cadena de caracteres		Secuencia de caracteres de código ASCII.
Numérico	Doble Real	
	Doble Complejo	
	Entero	
Booleano		<p>LabVIEW almacena datos Booleanos como valores de 8 bits. Son usados para representar un 1 o 0 (TRUE o FALSE respectivamente).</p> <p>Si el valor de 8 bits es cero, el valor Booleano es FALSE. Cualquier valor no igual a cero representa TRUE.</p>



Arreglo		Un arreglo es un conjunto de elementos del mismo tipo (Númerico, booleano, cadena o cluster) y puede ser de una o más dimensiones.
Clúster		Un clúster agrupa elementos de datos de diferentes tipos. Es equivalente a un registro o a una estructura en lenguajes de programación basados en texto.
Dinámico		Almacena información generada o adquirida por un VI Express (VIs que pueden ser configurados desde una ventana de diálogo) y se puede enlazar a cualquier indicador de tipo numérico, forma de onda o Booleano.

Tabla 5.1 . - Tipos de datos en LabVIEW

Cada uno de los bloques que pueden ser utilizados en el diagrama a bloques se encuentra en la paleta de funciones (Figura 5.7).

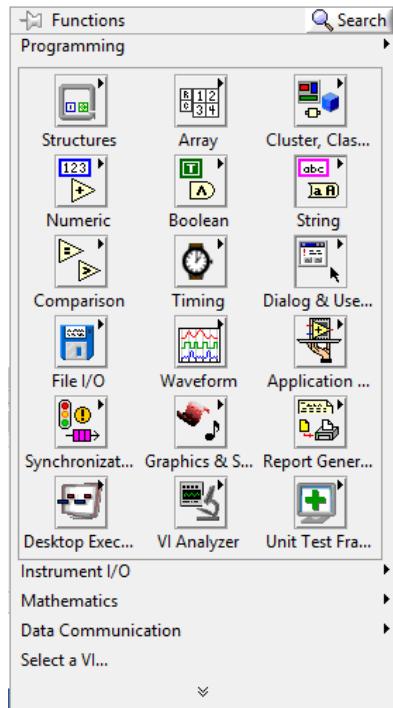


Figura 5.7 .- Paleta de funciones

Existe un gran número de bloques disponibles en la paleta de funciones, las funciones más destacadas son:

- Operaciones matemáticas (Figura 5.8): LabVIEW ofrece un amplio número de funciones matemáticas comunes como suma, resta, multiplicación, división, recíproco, valor absoluto, cuadrado, raíz cuadrada, suma de elementos de un arreglo, etc. además ofrece la posibilidad de capturar una expresión numérica con el número de entradas deseadas.

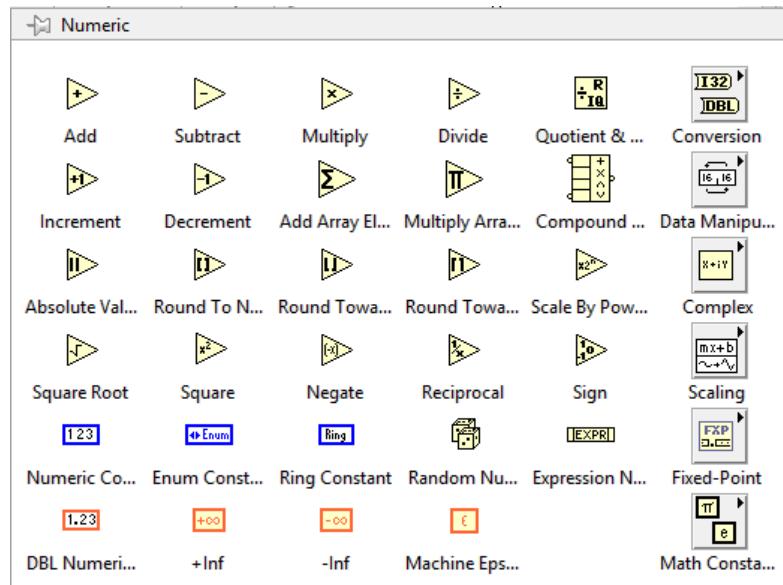


Figura 5.8 .- Funciones numéricas

- Operaciones booleanas y de comparación (Figura 5.9 y 5.10): Entre las operaciones booleanas disponibles en LabVIEW se encuentran las compuertas OR, AND, XOR, NOT, NAND, NOR, conversión de número a binario y viceversa, etc. Además se tienen operaciones con salidas booleanas (de comparación) como igual que, mayor que, menor que, diferente, etc.

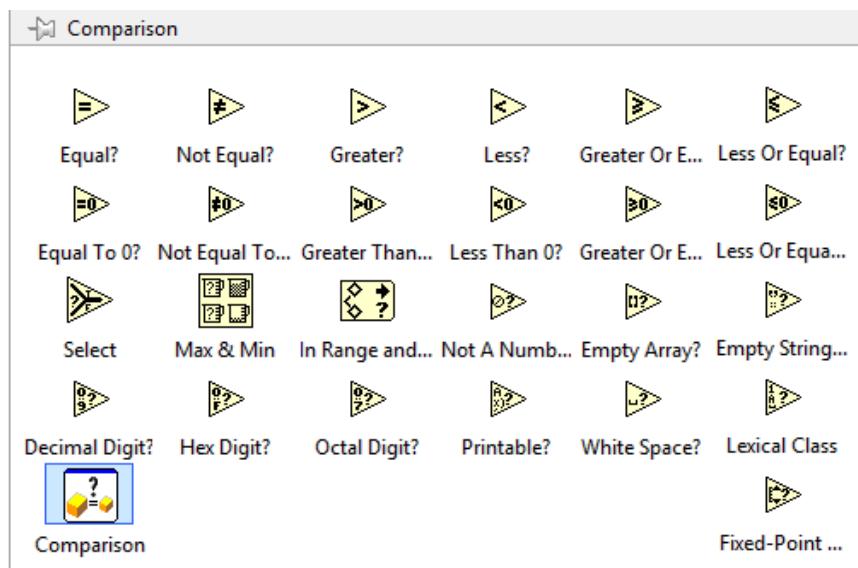
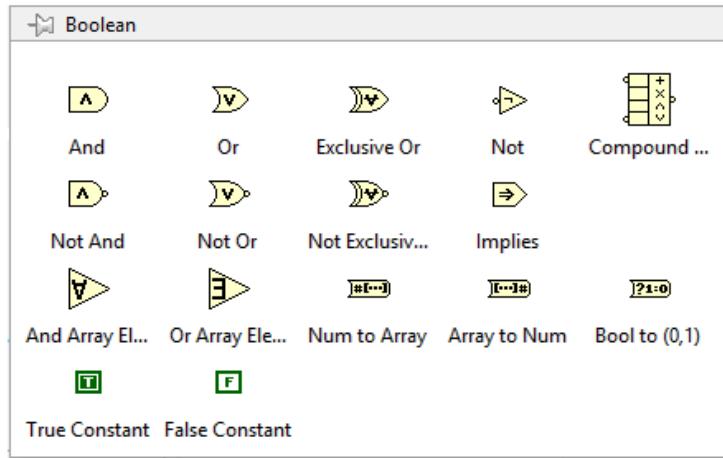
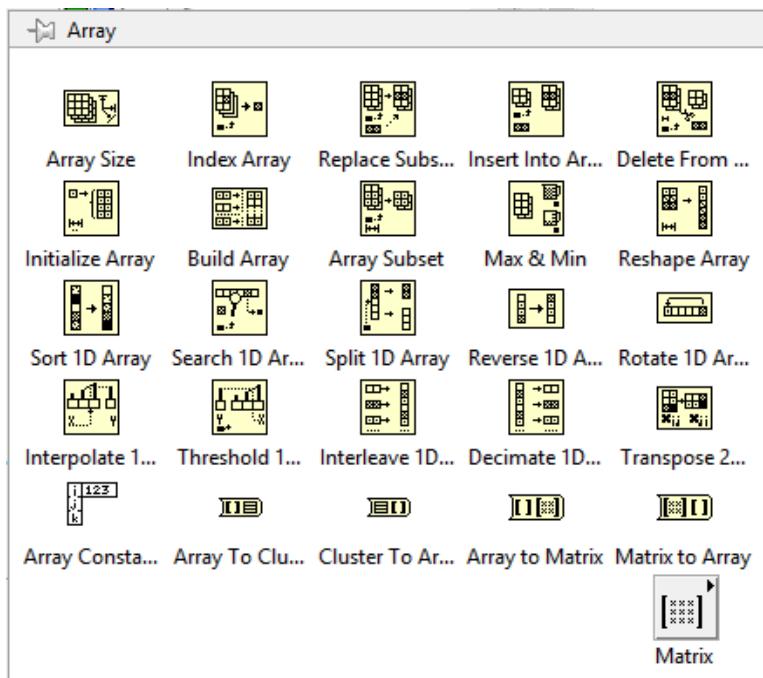


Figura 5.9 .- Funciones booleanas

**Figura 5.10** .- Funciones de comparación.

- Funciones con arreglos (Figura 5.11): LabVIEW contiene funciones muy específicas para trabajar con arreglos, como son, calcular el tamaño del arreglo, obtener el valor del elemento n de un arreglo, reemplazar elemento, insertar elemento, borrar elemento, inicializar arreglo, subconjunto de un arreglo, máximo y mínimo de un arreglo, separar, buscar, rotar, etc.

**Figura 5.11** .- Funciones con arreglos.



- Funciones con cadenas de caracteres (Figura 5.12): Las funciones que se pueden realizar con variables de tipo *string* pueden ser, entre otras, medir su tamaño, concatenar, obtener subconjunto, reemplazar, buscar y reemplazar, dar un formato específico, cambiar a mayúsculas o minúsculas, conversión a número y viceversa, etc.

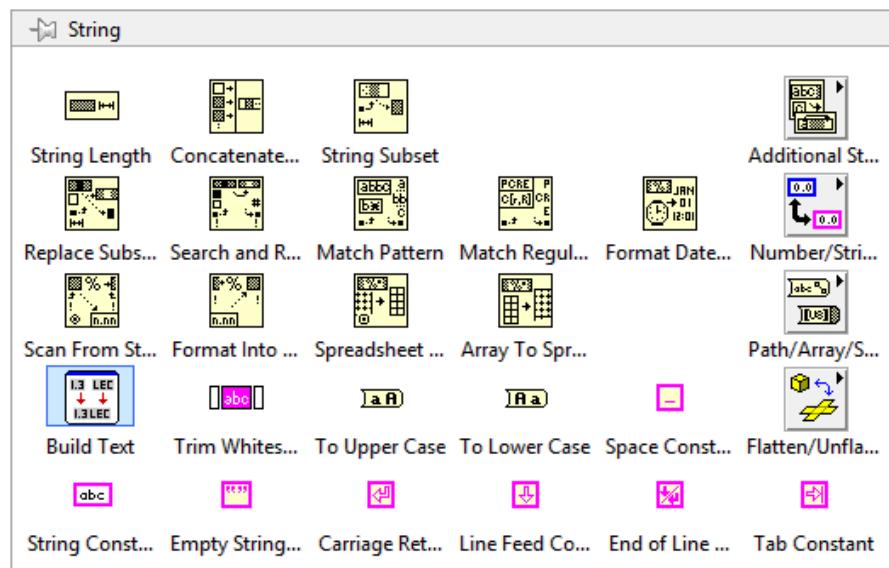


Figura 5.12 .- Funciones con cadenas de caracteres.

- Estructuras (Figura 5.13) : Una de las herramientas más comunes en un lenguaje de programación son los bucles, los cuales permiten repetir ciertas funciones por un cierto número de iteraciones si una condición se cumple. LabVIEW ofrece bloques para realizar las funciones de if, for, while, además permite programar eventos con un cierto orden en tiempo (flat sequence), líneas de código en lenguaje C (formula node), código en Matlab (Math Script), etc.

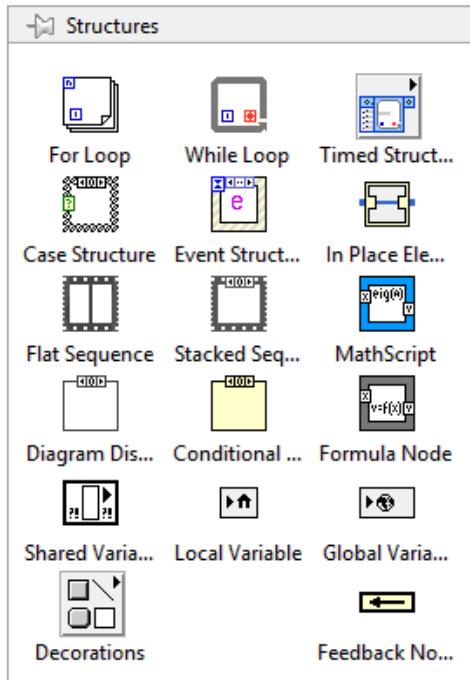


Figura 5.13 .- Estructuras en LabVIEW.

Icono y panel de conexión

El ícono es una representación visual de un VI y es necesario cuando en un Instrumento Virtual se requiere el uso de otro VI (sub VI), y la representación del subVI en el diagrama a bloques será la del ícono previamente seleccionado.

El panel de conexión identifica las entradas o salidas de un bloque (Figura 5.14).

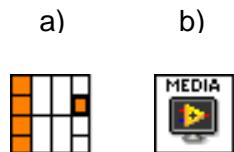


Figura 5.14 .- Panel de conexión (a) e Ícono de un VI (b) en LabVIEW



5.2 Programación del USRP N210 con LabVIEW

LabVIEW brinda la posibilidad de integrar distintas plataformas de hardware en un instrumento virtual, logrando así desarrollar instrumentos de medición sofisticados en tiempo real con los cuales se puede sacar provecho de todas las funciones del lenguaje LabVIEW para la captura, análisis y procesamiento de datos, tal es el caso del USRP N210.

Los bloques necesarios para recibir información proveniente del USRP N210 son:

- niUSRP Open Rx Session (Figura 5.15): Como su nombre lo indica, crea una sesión de recepción de datos proveniente del USRP, requiere la dirección IP del dispositivo como entrada y provee una salida para identificar y administrar la sesión de comunicación activa con el USRP (*session handle out*).

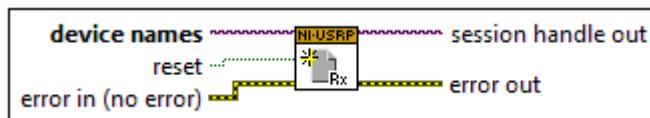


Figura 5.15 .- Bloque niUSRP Open Rx Session

- NiUSRP Configure Signal (Figura 5.16): Se encarga de configurar los parámetros necesarios para la señal de transmisión o de recepción, donde se debe especificar el canal y el nombre de la antena activa, la frecuencia de muestreo, frecuencia central y ganancia, además se debe ingresar la señal encargada del manejo de la sesión activa.

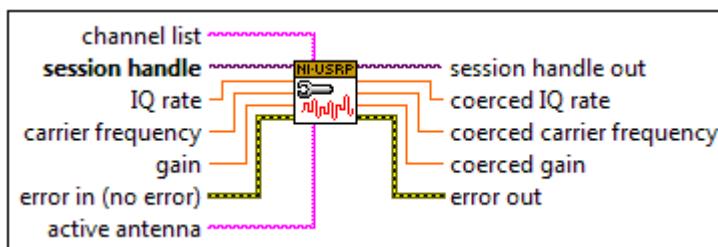


Figura 5.16 .- Bloque niUSRP Configure Signal



- niUSRP Initiate (Figura 5.17): Inicia la sesión de adquisición de datos, requiere la señal encargada del manejo de la sesión activa del USRP, el controlador del USRP pasa al estado de funcionamiento (*running state*).

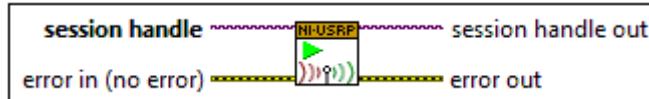


Figura 5.17 .- Bloque niUSRP Initiate

- niUSRP Fetch Rx Data (Figura 5.18): Entrega el número de muestras especificado que fueron obtenidas por el USRP provenientes del canal indicado. Dependiendo del número de canales activos los datos pueden ser de 1 o 2 dimensiones, además se puede seleccionar el tipo de datos de salida (Clúster, forma de onda, doble o entero)

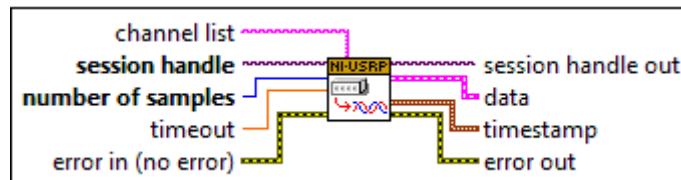


Figura 5.18 .- Bloque niUSRP Fetch Rx Data

- niUSRP Abort (Figura 5.19): Detiene la adquisición de datos iniciada previamente, el controlador del USRP pasa al estado terminado (*done state*)



Figura 5.19 .- Bloque niUSRP Abort

- niUSRP Close Session (Figura 5.20): Termina la sesión activa con el dispositivo



Figura 5.20 .- Bloque niUSRP Close Session



CAPÍTULO VI. DESARROLLO

Este capítulo presenta una descripción detallada de la implementación del receptor OFDM en una plataforma SDR usando el software LabVIEW. Se definen cada uno de los componentes del sistema, así como las funciones en software que conforman la capa PHY del receptor, basada en la norma IEEE 802.11g, las cuales permiten una correcta demodulación y decodificación de los símbolos OFDM recibidos.

6.1 Arquitectura del receptor OFDM

El receptor OFDM propuesto en éste trabajo está formado de dos componentes:

- **USRN N210 (Con tarjeta hija SBX):** Es la plataforma SDR con la que se desarrolló éste trabajo de tesis. El equipo USRP N210 de Ettus Research es una de las plataformas de hardware más comunes para la implementación de sistemas SDR. La tarjeta hija SBX trabaja en un rango de frecuencias de 400 – 4400 MHz. En la recepción el USRP permite la adquisición, filtrado de la señal de radiofrecuencia (Tarjeta hija), posteriormente las señales son convertidas a banda base (Tarjeta madre) y enviadas a una PC por medio de la interfaz Ethernet del USRP.

- **PC:** La función de la PC en el sistema de recepción OFDM es, en primer lugar definir las características del SDR tales como ganancia, ancho de banda y frecuencia central por medio del software LabVIEW. La segunda función de la PC es recibir los datos capturados en banda base, provenientes del USRP, para finalmente encargarse del procesamiento de la señal en banda base.

6.2 Procesamiento de la señal en banda base

6.2.1 Diagrama a bloques

La arquitectura general de la etapa de procesamiento de la señal en banda base del receptor OFDM implementado en el presente proyecto se muestra en la Figura 6.1.

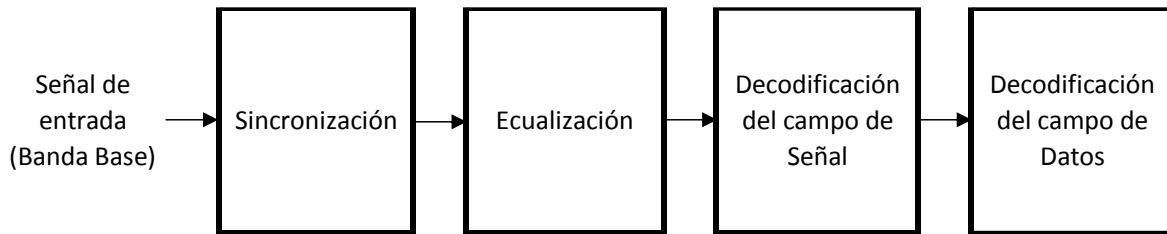


Figura 6.1 .- Diagrama a bloques del receptor OFDM

Como se puede observar en el diagrama a bloques, la fase de recepción propuesta en éste trabajo de tesis está formada por 4 etapas principales, cada etapa a su vez, se encuentra formada por otras funciones, cada una de las cuales serán descritas en las siguientes secciones.

6.2.2 Etapa de sincronización

La etapa de sincronización está formada por funciones relacionadas con el procesamiento de las secuencias de entrenamiento enviadas al inicio de cada trama OFDM (preámbulo PLCP), la tarea de la etapa de sincronización es identificar y delimitar una trama OFDM, además de corregir (en frecuencia y tiempo) los símbolos OFDM usando las propiedades periódicas del preámbulo PLCP. Las funciones que conforman la etapa de sincronización se muestran en la Figura 6.2.

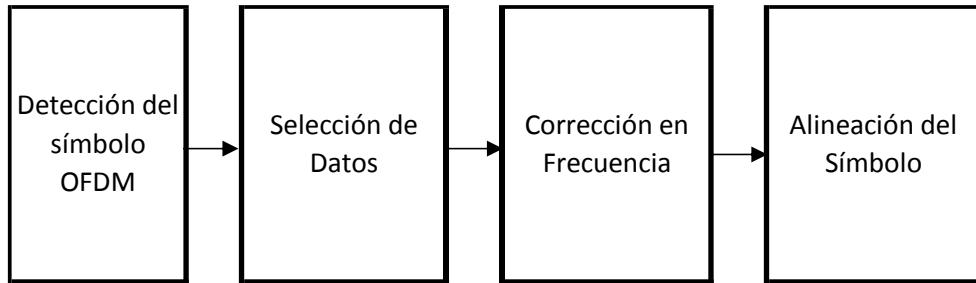


Figura 6.2 .. Bloques que conforman la etapa de sincronización

Detección del Símbolo: La tarea del bloque de detección es identificar el inicio de un frame OFDM con la ayuda del preámbulo PLCP, en particular por medio de las secuencias de entrenamiento cortas enviadas al inicio del frame OFDM.

El principio de funcionamiento del bloque de detección del frame forma parte de otro trabajo de tesis complementario que también cubre la implementación de un transmisor OFDM, por lo cual el presente proyecto de tesis abarca apartir del bloque de corrección en frecuencia hasta la decodificación del campo de datos de la trama recibida.

Selección de Datos: Debido a que en éste punto no se sabe la longitud de la trama OFDM que fue enviada, es decir, se desconoce el número de muestras que corresponden a dicha trama, la función del selector de datos es tomar aquellas muestras que pertenezcan a la trama OFDM y despreciar aquellas que no, con el objetivo de evitar procesamiento innecesario en las etapas posteriores en donde aún se desconoce la longitud de la trama enviada.



Si un gran número de muestras consecutivas están por debajo de un umbral específico, se considera que dichas muestras no forman parte de la trama OFDM. La implementación de la etapa de selección de datos se muestra en Anexo A.

Corrección en frecuencia: Una vez que la trama OFDM es recibida se debe hacer una corrección en frecuencia debido a que los osciladores locales del transmisor y receptor pueden tener una variación en frecuencia entre sí. Para estimar el offset en frecuencia se usa el algoritmo sugerido en [30], en donde se emplean las últimas 5 secuencias de entrenamiento cortas.

Cada secuencia de entrenamiento está formada de 16 muestras repetidas periódicamente, por lo que se requieren 80 muestras en total para realizar la corrección. Idealmente cada muestra $s[n]$ debe ser igual a la muestra $s[n+16]$ por lo que:

$$(6-1) \quad s[n] * \bar{s}[n + 16] \in \mathbb{R}$$

Sin embargo debido a la variación del oscilador entre el transmisor y el receptor, el producto de la ecuación (6-1) es un número complejo y su argumento es proporcional a la rotación introducida por la variación en frecuencia, como lo muestra la ecuación (6-2).

$$(6-2) \quad \Delta f = \frac{1}{16} \angle \sum_{n=0}^{63} s[n] * \bar{s}[n + 16]$$

La programación del bloque de corrección en frecuencia se muestra en la Figura 6.3.

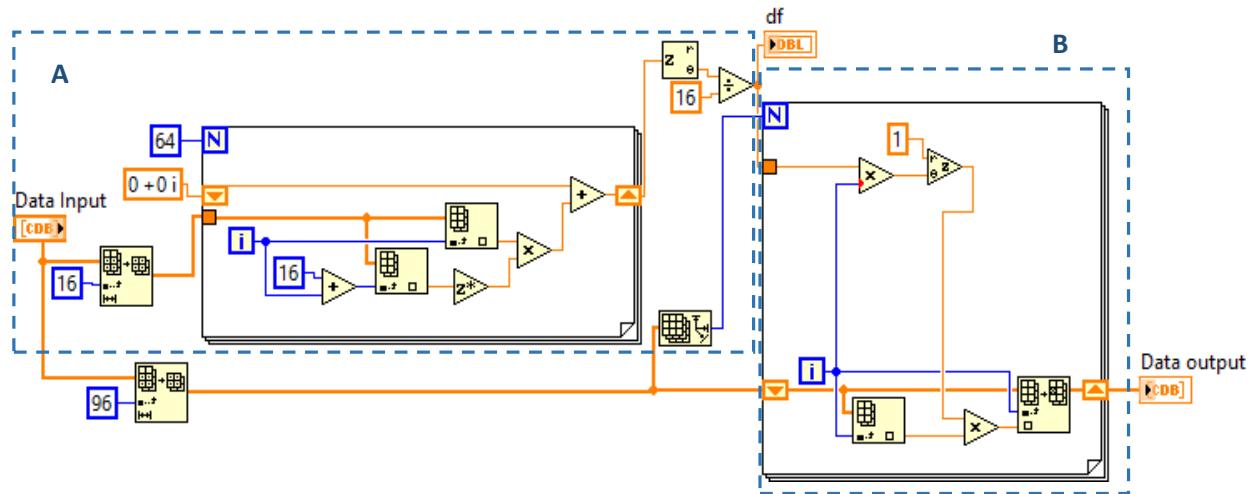


Figura 6.3 .- Implementación del bloque de corrección en frecuencia

Como se puede observar en la Figura 6.3, la implementación está dividida en dos etapas, en la primera parte (A) se realiza el cálculo del offset usando las últimas 5 secuencias cortas por medio de un bucle for para realizar la suma de cada producto y finalmente se obtiene el ángulo y se divide entre 16, como se muestra en la ecuación (6-2).

Finalmente en la segunda etapa (B) se aplica la corrección para cada muestra de entrada, usando la ecuación (6-3), donde cada muestra corregida se representa con $s'[n]$. De igual forma se usa un bucle for para recorrer todas las muestras, adicionalmente se usa un registro de corrimiento que permite sustituir el valor de una muestra por su valor corregido.

$$(6-3) \quad s'[n] = s[n]e^{i n df}$$



Alineación del símbolo: El bloque de alineación del símbolo usa las secuencias de entrenamiento largas, el objetivo es identificar el comienzo exacto de la trama OFDM, extraer cada símbolo OFDM y remover el prefijo cíclico.

El algoritmo usado se basa en el implementado en [15], en donde se calcula la correlación entre el valor conocido de una secuencia de entrenamiento larga con los datos de entrada.

Debido a que como parte del preámbulo se envían dos secuencias de entrenamiento, más el ciclo prefijo (como se mostró en la Figura 3.12), el resultado de la correlación tendrá 3 valores máximos, cada uno de los cuales corresponden al inicio de cada secuencia de entrenamiento y al inicio del ciclo prefijo. Para determinar el número de muestra que coincide con el inicio de la última secuencia de entrenamiento, se debe tomar el último valor (número de muestra mayor) de entre los 3 valores máximos de correlación.

Por lo tanto, si N_p es el arreglo que contiene los 3 números de muestra ‘n’ que maximizan la correlación entre la secuencia de entrenamiento (LTS) y las muestras de entrada $s(n)$, N_p está dado por la ecuación (6-4).

$$(6-4) \quad N_p = \arg \max_3 \sum_{k=0}^{63} s[n+k] * \overline{LTS}[k]$$

Nótese que la correlación es calculada por 64 muestras, que es la duración de la secuencia de entrenamiento larga.

El número de muestra que corresponde con el inicio de la última secuencia de entrenamiento larga es el valor máximo de N_p y entonces, el inicio del primer símbolo OFDM n_p está dado por la ecuación 6-5, en donde se puede observar que al valor máximo de N_p se le suman 64, que es la longitud de la secuencia de entrenamiento larga.

$$(6-5) \quad n_p = \max(N_p) + 64$$

La implementación de la alineación del símbolo se muestra en la Figura 6.4.

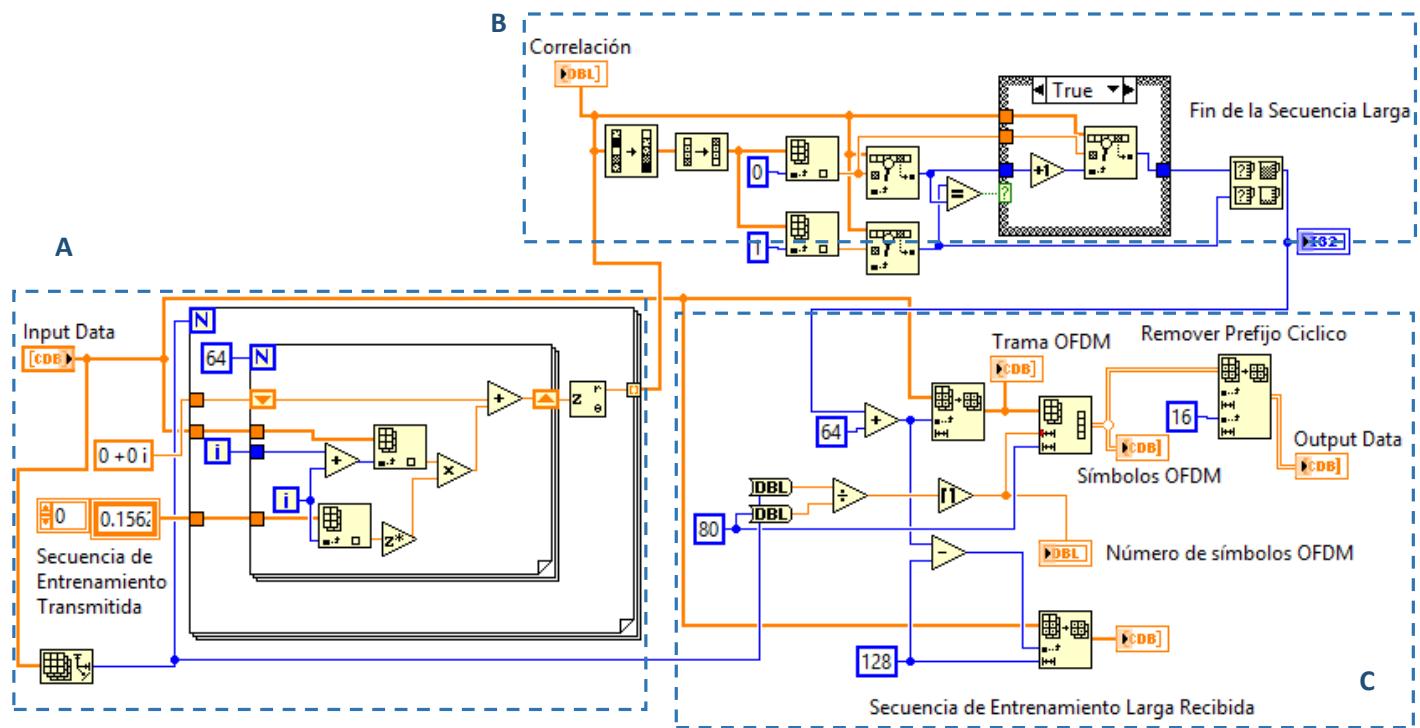


Figura 6.4 .- Implementación del bloque de alineación del símbolo

Como se muestra en la Figura 6.4, el bloque de alineación del símbolo está formado por 3 etapas principales, en la etapa A se realiza el cálculo de la correlación entre la secuencia de entrenamiento y las muestras de entrada de acuerdo a la ecuación 6-4.



En la etapa B se encuentra los números de muestras que maximizan la correlación, para esto, los valores de correlación se ordenan de mayor a menor y posteriormente para los dos valores más altos se encuentra su número de muestra respectivo, se toma el mayor de ellos y se le suma 64 para obtener el número de muestra del primer símbolo OFDM.

Finalmente en la etapa C, se construye cada símbolo OFDM inicialmente con 80 muestras cada uno, lo cual incluye 64 muestras de datos y 16 muestras de prefijo cíclico, para finalmente obtener 64 muestras al remover el ciclo prefijo.

6.2.3 Etapa de ecualización

La etapa de ecualización es la encargada de procesar los datos recibidos, identificar los efectos de la selectividad de frecuencias del medio (Estimación del canal) y contrarrestar su efecto en los símbolos OFDM recibidos (Ecualización y Corrección en fase). Los bloques que forman la etapa de ecualización se muestran en la Figura 6.5.

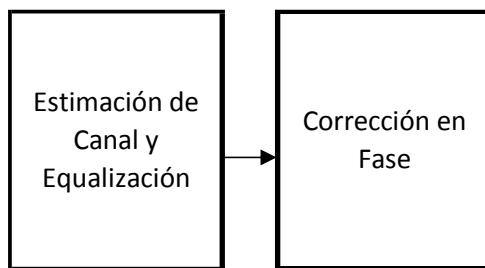


Figura 6.5 .- Diagrama a bloques de la etapa de ecualización

Estimador de canal y ecualización: La estimación del canal permite conocer la atenuación de cada portadora al ser transmitidas por el medio inalámbrico. Para lograr determinar dicha atenuación, es necesario conocer el valor transmitido en cada portadora (portadoras piloto) para poder compararlo con su valor recibido respectivo, como se muestra en la Figura 6.6.

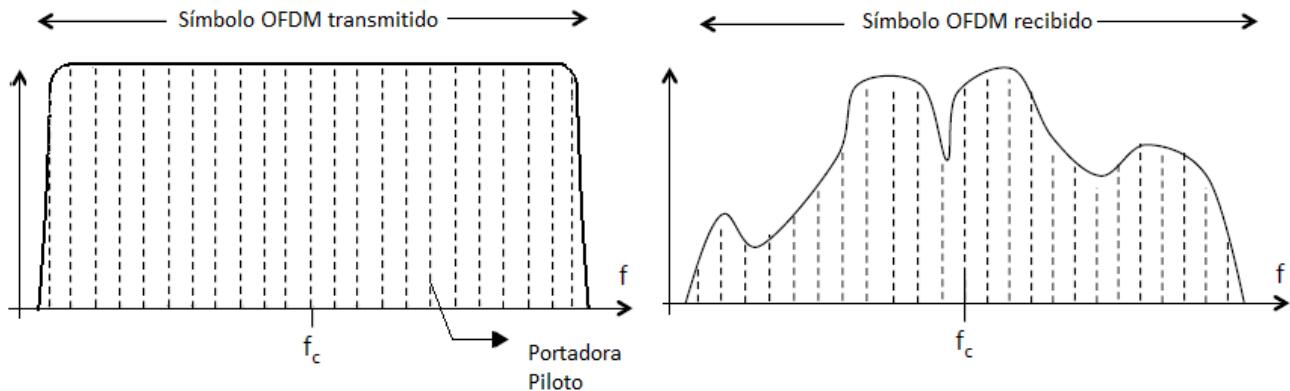


Figura 6.6 .- Efecto de un canal selectivo en frecuencia

Debido a que se requiere el envío de un valor piloto en cada portadora, se usaron ambas secuencias de entrenamiento largas para la estimación del canal, una vez conocido el efecto del medio en cada portadora, el ecualizador permite escalar las portadoras para eliminar la atenuación introducida por el canal. El tipo de ecualización implementada en éste trabajo de tesis es la del ecualizador ‘one-tap’ descrito en [31].

Sea X_n el valor de la secuencia de entrenamiento larga conocida en la portadora n y L_n el promedio de ambos valores de las secuencias recibidas en esa portadora, la atenuación λ_n ó estimador de cada canal está dada por la ecuación 6-6.

$$(6-6) \quad \lambda_n = L_n * \frac{\overline{X_n}}{|X_n|^2} \quad \text{para } n = 1, \dots, 64$$

Finalmente para ecualizar cada una de las portadoras y quitar la atenuación λ_n se aplica la ecuación (6-7) a cada muestra recibida S , donde S'_n es el valor de la portadora n corregida.

$$(6-7) \quad S'_n = S_n * \frac{\overline{\lambda_n}}{|\lambda_n|^2} \quad \text{para } n = 1, \dots, 64$$

La implementación de la etapa de estimación de canal y ecualización se muestra en la Figura 6.7.

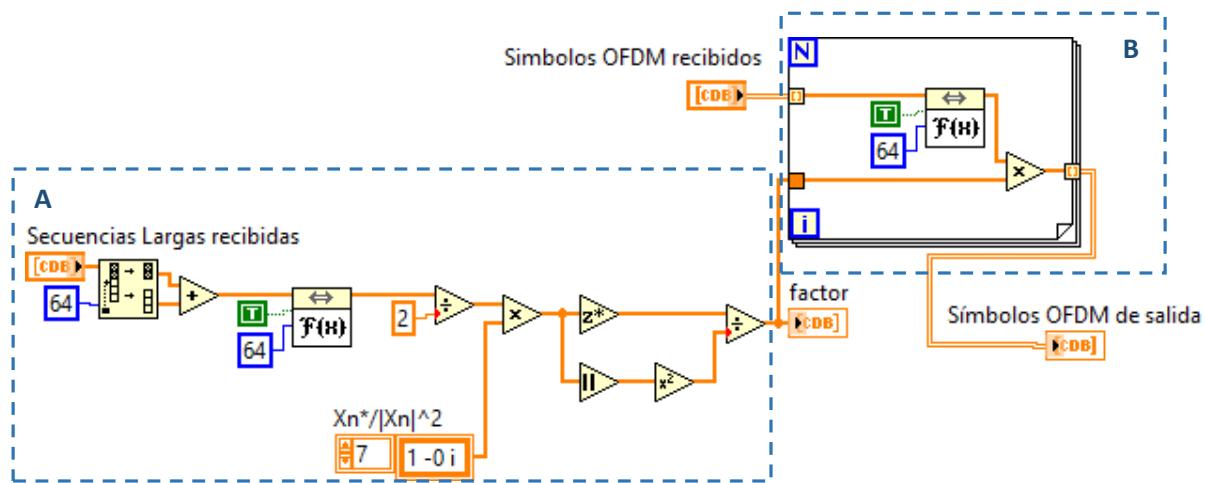


Figura 6.7 .- Implementación de estimación de canal y ecualización

La etapa A de la implementación mostrada en la Figura 6.7 corresponde a la estimación del canal, en donde se obtiene el estimador λ para cada portadora, nótese que las secuencias de entrenamiento largas recibidas deben transformarse al dominio de la frecuencia por medio de la transformada FFT, además el factor que multiplica la secuencia de entrenamiento larga recibida de la ecuación 6-6 se trata de un valor constante.

Una vez calculado el estimador λ se aplica la ecualización para las muestras de entrada, de igual forma se debe aplicar la transformada FFT para convertirlas al dominio de la frecuencia y así aplicar el estimador a cada portadora por medio de un ciclo for.

Corrección de fase: Debido a que la estimación de canal del bloque anterior se realiza al inicio de la trama OFDM (en el preámbulo PLCP), es necesario realizar una corrección en fase individual de cada símbolo OFDM, con el objetivo de asegurar una ecualización precisa tomando en consideración las portadoras piloto que se insertan en cada símbolo OFDM, como se ilustra en la Figura 6.8.

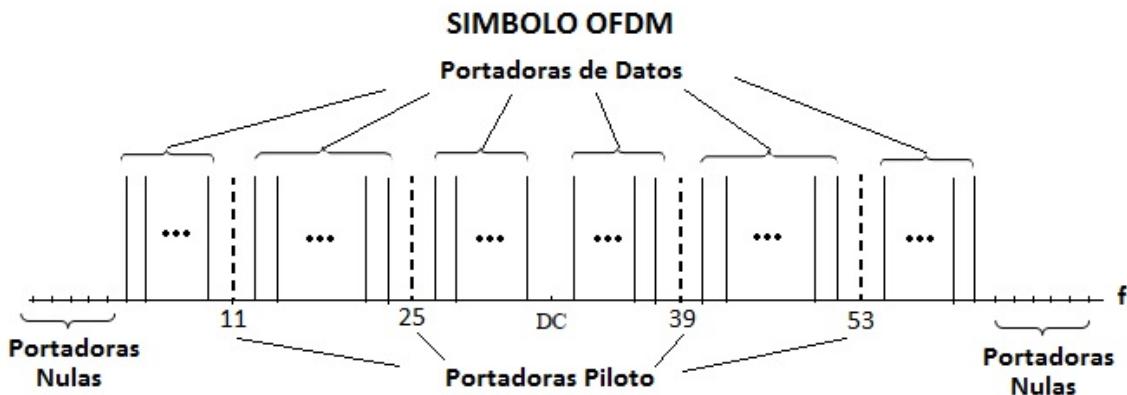


Figura 6.8 .- Distribución de las portadoras en un símbolo OFDM

El valor de las portadoras piloto transmitidas es un número real, conforme lo establece el estándar, sin embargo al existir algún desfasamiento, el valor de las portadoras piloto recibidas se convierte en un número complejo y en consecuencia su fase es distinta de cero. Por lo tanto, la fase de cada portadora piloto recibida representa un desfasamiento y para estimar dicho desfasamiento de las portadoras de datos se usa una regresión lineal [32].

Sea $y = \{y_1, y_2, y_3, y_4\}$ la fase de cada piloto cuya posición respectiva es $x = \{x_1, x_2, x_3, x_4\}$ dentro de un símbolo OFDM, la ecuación de la línea recta para la regresión lineal es (ecuación 6-8):

$$(6-8) \quad Y = \alpha + \beta X$$

Donde α y β reciben el nombre de coeficientes de regresión y están dados por las ecuaciones 6-9 y 6-10.

$$(6-9) \quad \beta = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{cov(x, y)}{var(x)} \quad \text{para } n = 1, \dots, 4$$

$$(6-10) \quad \alpha = \bar{y} - \beta \bar{x}$$



Finalmente el desfasamiento estimado de cada muestra de datos de entrada $s[n]$ se puede corregir mediante la ecuación 6-11.

$$(6-11) \quad s'[n] = s[n] * e^{-jY} = s[n] * e^{-(\alpha+\beta n)j}$$

Una vez que los datos han sido ecualizados, las portadoras piloto no son necesarias, por lo que se extraen de cada símbolo OFDM, así como las portadoras nulas, quedando cada símbolo OFDM conformado de 48 portadoras de datos.

La implementación en el software LabVIEW de la etapa de corrección en fase se muestra en el Anexo B.

6.2.4 Decodificación del campo de señal

Tal como lo describe el estándar IEEE 802.11, el primer símbolo OFDM corresponde al campo de señal, el cual contiene la información necesaria para recuperar los datos enviados, tal como el tipo de codificación y la longitud del mensaje, por lo cual es necesario en primer lugar recuperar el campo de señal para posteriormente decodificar los datos.

La decodificación del campo de señal está formada por 3 etapas, las cuales se muestran en la Figura 6.9.



Figura 6.9 .- Diagrama a bloques de la etapa de decodificación



Demodulación Señal: La función del bloque de demodulación es realizar el mapeo de los símbolos BPSK del campo de señal a bits, su implementación en LabVIEW se muestra en la Figura 6.10.

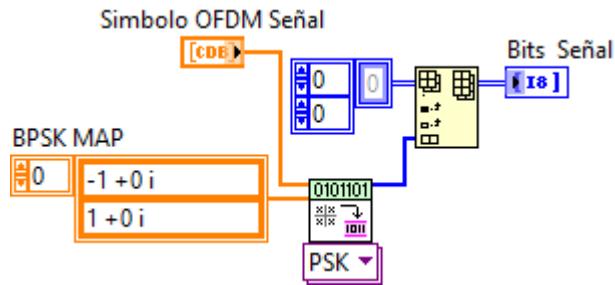


Figura 6.10 .- Demodulación del campo de Señal

La demodulación se lleva a cabo mediante el VI llamado *Map Symbols to bits.vi*, el cual requiere que se indique el tipo de modulación de los datos de entrada (PSK, QAM, etc) y el mapa de símbolos que representa la correspondencia entre valores del símbolo (puntos de la constelación) y bits.

Deinterleaver: Como su nombre lo indica, el deinterleaver realiza exactamente la operación inversa de su contraparte en el transmisor. Como se mencionó en el estudio del estándar IEEE 802.11, el interleaver está formado por dos permutaciones que definen nuevas posiciones a los bits de entrada, por lo tanto en la recepción, los bits deben regresar a sus posiciones originales, la función de deinterleaver debe realizarse en bloques de N_{CBPS} bits.

Las permutaciones realizadas en el deinterleaver están de igual forma en función de la posición de los bits recibidos y del número de bits codificados por símbolo OFDM (N_{CBPS}) que a su vez depende del tipo de modulación y se encuentran especificados en la tabla 3.2.



La nueva posición de cada bit después de la primera permutación en el proceso del deinterleaver está representada por i en la ecuación (6-12), mientras que la posición después de la segunda permutación y por tanto, la posición original en la que fueron enviados los bits, está representada por k de la ecuación (6-13).

$$(6-12) \quad i = s * \left\lfloor \frac{i}{s} \right\rfloor + \left(i + \left\lfloor 16 * \frac{i}{N_{CBPS}} \right\rfloor \right) \bmod s \quad i = 0, 1, \dots, N_{CBPS} - 1$$

$$(6-13) \quad k = 16i - (N_{CBPS} - 1)[16 * i / N_{CBPS}] \quad i = 0, 1, \dots, N_{CBPS} - 1$$

Donde: $s = \max(N_{BPSC}/2, 1)$

Para el caso del campo de señal, el cual esta codificado con modulación BPSK 1/2, el valor de N_{CBPS} es igual a 48 y s es igual a 1.

Una vez que los bits han regresado a su posición original, los bits de datos provenientes de cada símbolo OFDM se unen para formar una sola cadena de bits de información. La implementación de la función de deinterleaver se encuentra en el Anexo C.

Decodificador: La parte de decodificación convolucional se realiza usando un decodificador de Viterbi, el cual realiza la decodificación por medio de una estructura de Trellis para reconstruir los datos enviados originalmente.

El VI que se usa para la decodificación es *Convolutional Decode.vi*, al que se le debe especificar los polinomios generadores, la longitud de restricción (K) y la profundidad de búsqueda (traceback depth, D) al realizar la decodificación.

La profundidad de búsqueda (D) indica el número de ramales de la estructura de Trellis a los que se puede acceder al realizar la decodificación. Por lo general es 5 veces la longitud de restricción. Un número pequeño de profundidad de búsqueda implicaría un gran número de errores, aunque un número grande exigiría gran cantidad de procesamiento de información, lo que repercute directamente en la velocidad del sistema.

La implementación de la etapa de decodificación se muestra en la Figura 6.11, una vez decodificados los bits del campo de señal, es posible extraer los parámetros buscados (tasa de codificación de los datos y longitud).

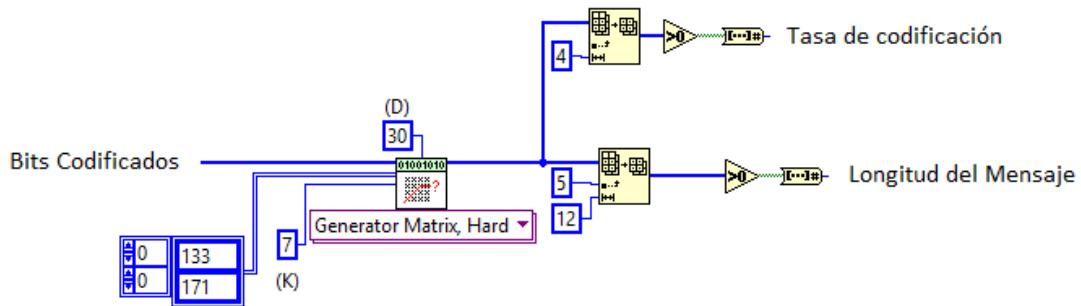


Figura 6.11 .- Implementación de la etapa de decodificación

6.2.5 Decodificación del campo de datos

Las etapas necesarias para la correcta decodificación de la información contenida en el campo de datos se muestran en la Figura 6.12.

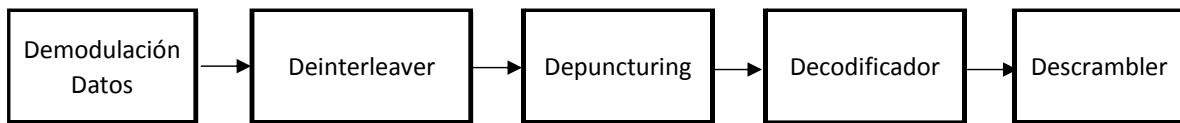


Figura 6.12 .- Decodificación de la información



Demodulación Datos: La Demodulación de datos se realiza de forma similar a la demodulación del campo de señal de la sección anterior, en éste bloque se realiza el mapeo de los símbolos de datos BPSK, QPSK 16-QAM o 64-QAM según corresponda a bits, su implementación en LabVIEW se muestra en el Anexo D.

La demodulación se lleva a cabo mediante una estructura tipo *case*, en la cual se debe indicar el tipo de modulación de los datos mediante un número entero (0 para BPSK, 1 para QPSK, 2 para 16-QAM y 3 para el caso de 64-QAM) y dependiendo la modulación, el VI *Map Symbols to bits.vi* recibe los parámetros de configuración necesarios antes mencionados (tipo de modulación PSK o QAM y el mapa de símbolos).

Deinterleaver: La etapa de interleaver para el campo de datos es idéntica a la explicada en la sección del campo de señal, en donde se tienen dos permutaciones para reordenar los bits recibidos.

Depuncturing: El objetivo del proceso de *depuncturing* es insertar los bits que se quitaron en la etapa de acortamiento en el transmisor.

Debido a que evidentemente los valores de esos bits son desconocidos para el receptor, en esas posiciones se insertan valores de '0' para que el codificador convolucional corrija esos bits.

Los bits recibidos que tienen valor de '0' lógico se sustituyen con valor de '-1' y los bits con valor de '1' permanecen así, de tal forma que los bits perdidos en el proceso de acortamiento toman un valor neutro para que sea el decodificador convolucional el que determine su valor. La implementación del proceso de Depuncturing se muestra en el Anexo E.



Decodificador: Es la misma etapa que la del decodificador convolucional usado para el campo de señal, con la única diferencia de que en este decodificador se debe especificar que los valores de los datos de entrada van de -1 a 1, eso se logra con la opción *UnQuantized*, como lo muestra la Figura 6.13

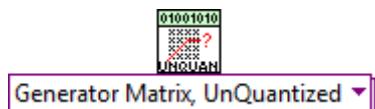


Figura 6.13 .- VI usado para la decodificación convolucional de los datos

Descrambler: El mismo algoritmo de aleatorización (usado en el transmisor) permite realizar la función inversa (descrambler) en el receptor.

Se usa un registro de corrimiento y aritmética módulo 2 para determinar el bit de salida a partir del bit de entrada y de las posiciones específicas del registro de corrimiento, dichas posiciones están definidas por el polinomio generalizador x^7+x^4+1 definido en el estándar IEEE 802.11.

La implementación en LabVIEW de la etapa del descrambler se muestra en el Anexo F de éste trabajo de tesis.

Posterior a la etapa de descrambler se debe comprobar que la trama recibida está libre de errores mediante el código de redundancia cíclica. Se calcula el campo FCS de los datos recibidos, mediante el VI CRC-32.vi y se compara con el campo FCS recibido, si ambos campos son iguales, entonces se asume que la trama recibida es correcta y por lo tanto, no contiene errores.



CAPÍTULO VII. RESULTADOS

En éste capítulo se presentan las pruebas realizadas para evaluar el desempeño del receptor OFDM propuesto en éste trabajo de tesis.

El presente capítulo está dividido en dos secciones: la primera sección muestra las pruebas realizadas para comprobar la implementación y operación de los bloques que llevan a cabo el procesamiento de la señal en banda base más relevantes (hasta la etapa de demodulación) y los cuales fueron implementados en el software LabVIEW.

La segunda sección pretende mostrar los resultados obtenidos al implementar un sistema de comunicación real entre un transmisor y un receptor OFDM, se muestran estadísticas tales como el número de bits perdidos (BER) en función de la distancia entre transmisor y receptor, así como del BER en función del SNR para distintos escenarios.

7.1 Comprobación de los bloques de procesamiento en banda base

7.1.1 Selector de Datos

Como se mencionó en el capítulo anterior, el bloque de Selector de Datos permite filtrar el número de muestras que son enviadas a los bloques posteriores para evitar que dichos bloques procesen muestras que no pertenecen a la trama OFDM.

La Figura 7.1 muestra los datos recibidos después de que el bloque de Detección del símbolo OFDM ha detectado el inicio de una trama. Como se puede observar, en este caso, el número de muestras que contienen información son alrededor de 4000 muestras de las 44000 muestras capturadas por el USRP.

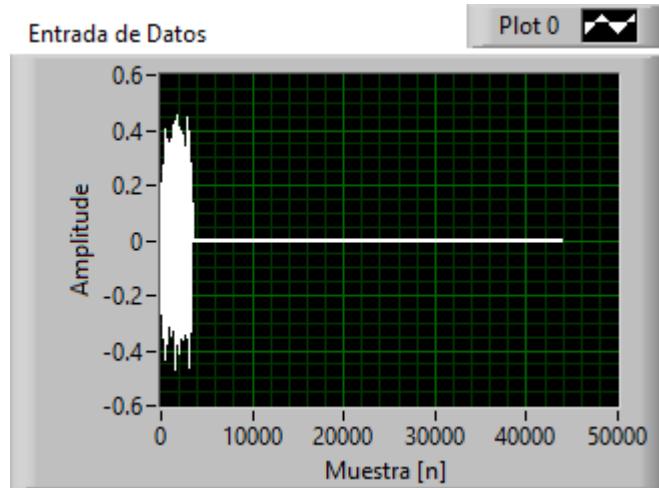


Figura 7.1 .- Entrada de datos provenientes del USRP

Una vez que los datos son procesados por el algoritmo de selección de datos, las muestras útiles que pasarán a los siguientes bloques se muestran en la Figura 7.2.

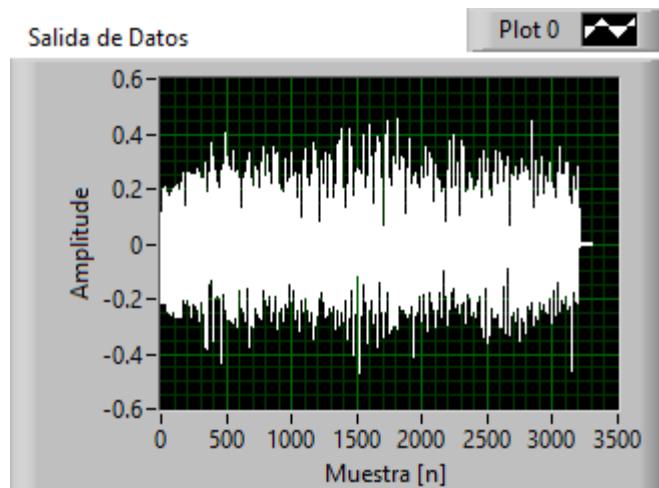


Figura 7.2 .- Salida del bloque de Selección de Datos

7.1.2 Alineación del símbolo

Una de las funciones críticas en la recepción de la trama OFDM es la de identificar exactamente la muestra de inicio de los símbolos de datos OFDM, dicha función es realizada por el bloque de alineación del símbolo.

Se realiza la correlación de las muestras de entrada con la secuencia de entrenamiento larga conocida y debido a que en el preámbulo PLCP de la trama OFDM se incluye dicha secuencia 2.5 veces (2 secuencias más ciclo prefijo) el resultado de la correlación debe contener 3 valores máximos.

El valor de la correlación calculada en el bloque de alineación del símbolo OFDM se muestra en la Figura 7.3. De los 3 picos de correlación, se debe identificar el valor máximo que corresponda al número de muestra mayor, dicho valor representa la posición de la última secuencia de entrenamiento larga.

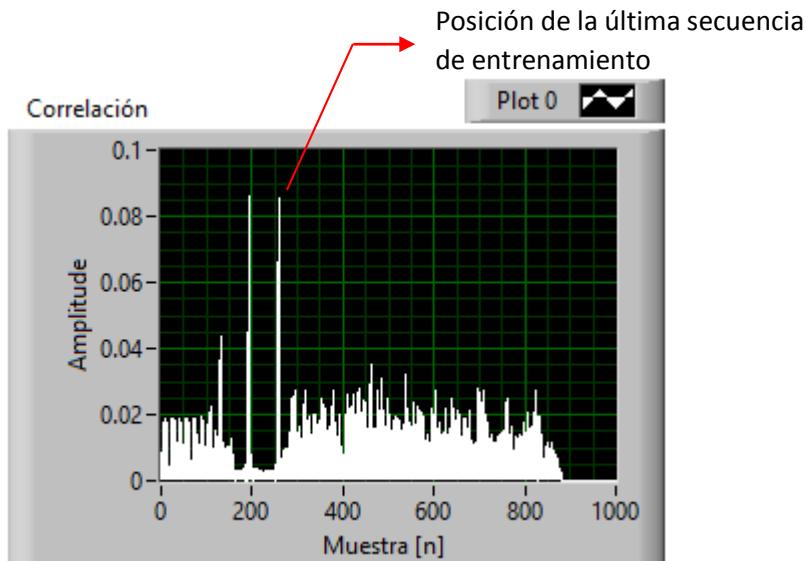


Figura 7.3 .- Correlación entre muestras de entrada y secuencia de entrenamiento larga

Una vez obtenida la posición de la primera muestra perteneciente a la última secuencia de entrenamiento, como la indicada en la Figura 7.2, el bloque de alineación del símbolo, suma dicha posición más la longitud de la secuencia de entrenamiento (64 muestras), obteniendo así la posición del primer símbolo OFDM.

7.1.3 Estimación del canal y ecualización

Éste bloque es el encargado de compensar la atenuación que sufre cada portadora al ser enviada por el medio inalámbrico, o bien causada por el propio USRP en la etapa de filtrado de la señal de radiofrecuencia. En la etapa de estimación del canal se usan las secuencias de entrenamiento largas, las cuales tienen un valor conocido para estimar el canal.

La Figura 7.4 muestra un ejemplo de la etapa de estimación del canal de una trama OFDM capturada con el USRP. Idealmente en un ambiente sin ruido, sin multirayos y con una respuesta ideal de los filtros del USRP, el valor de la estimación del canal debe ser constante a lo largo de las subportadoras.

Nótese que los valores de cada portadora de la Figura 7.4 corresponden a la secuencia de entrenamiento larga recibida.

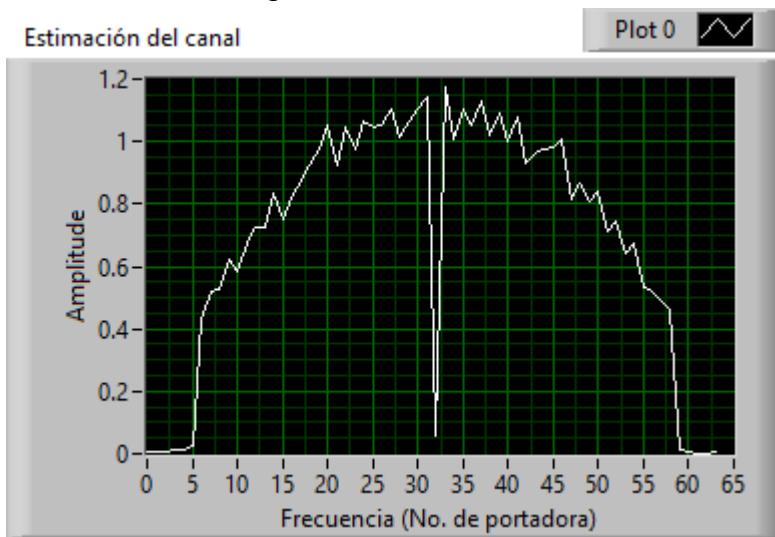


Figura 7.4 .- Ejemplo de la estimación del canal de una trama OFDM recibida

De la Figura 7.4 se puede confirmar la presencia de las portadoras nulas (portadoras 0-5, 32 y 59-64), pero también se observa que las portadoras ubicadas a los extremos del ancho de banda sufren una mayor atenuación y dicha atenuación va disminuyendo conforme se acercan a la portadora de DC, la cual corresponde a la frecuencia central.

Una vez calculado el factor que compensa la atenuación de cada portadora de acuerdo a las ecuaciones 6-6 y 6-7, las muestras de entrada son corregidas.

Las portadoras de la secuencia de entrenamiento (portadoras piloto) corregidas, se muestran en la Figura 7.5.

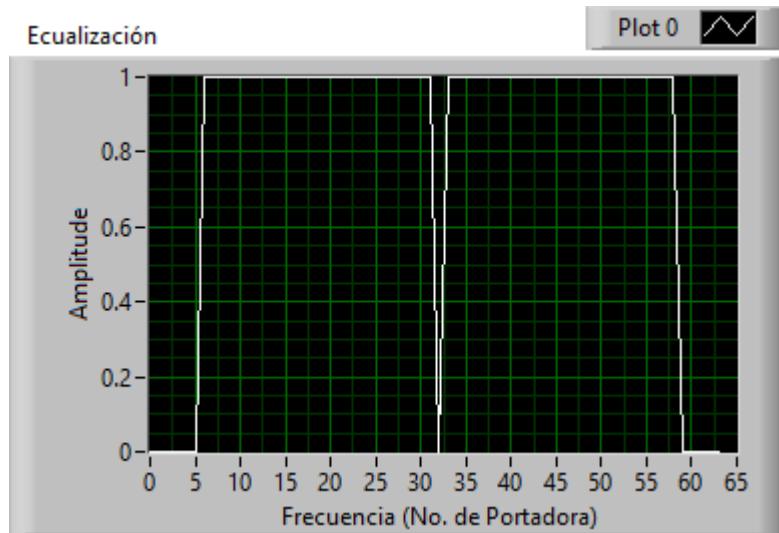


Figura 7.5 .- Portadoras piloto después de la etapa de Ecualización

El efecto de la respuesta no homogénea del canal a cada portadora sobre los símbolos de datos (con modulación PSK y QAM) contenidos en el símbolo OFDM se muestra del lado izquierdo de las Figuras 7.6, 7.7 y 7.8. Como se puede observar, las constelaciones recibidas (antes del bloque de estimación del canal y ecualización) se encuentran dispersas.

Por otro lado, el efecto que tiene el uso de la ecualización en los datos con modulación BPSK, QPSK y 16-QAM también se muestra, en el lado derecho de la Figuras 7.6, 7.7 y 7.8 respectivamente.

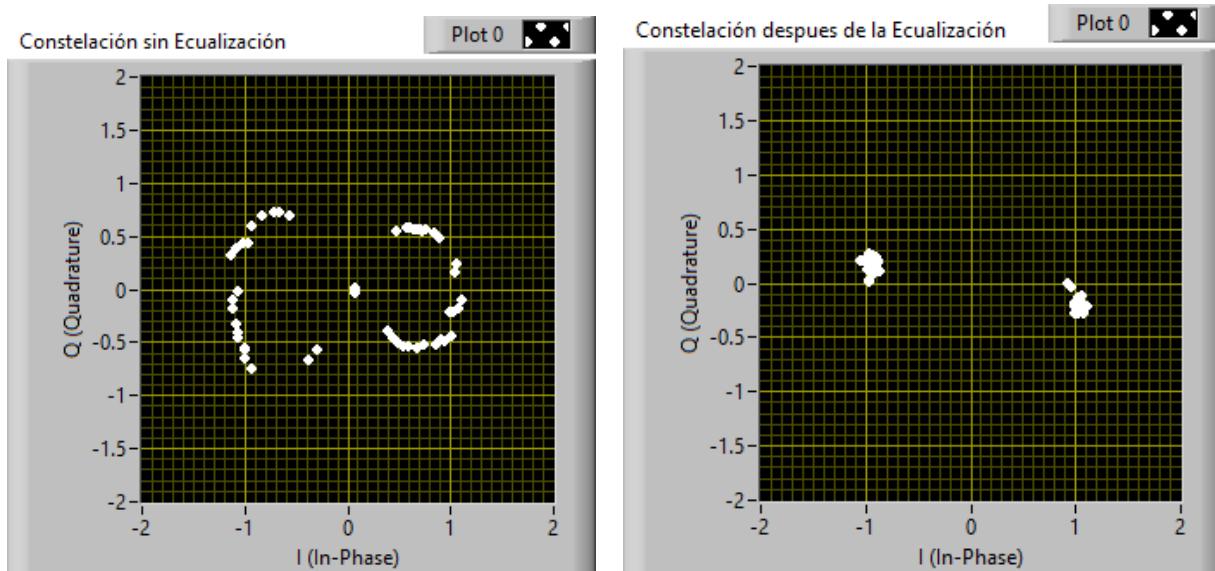


Figura 7.6 .- Comparación de la constelación BPSK sin ecualización (izquierda) y con ecualización (derecha)

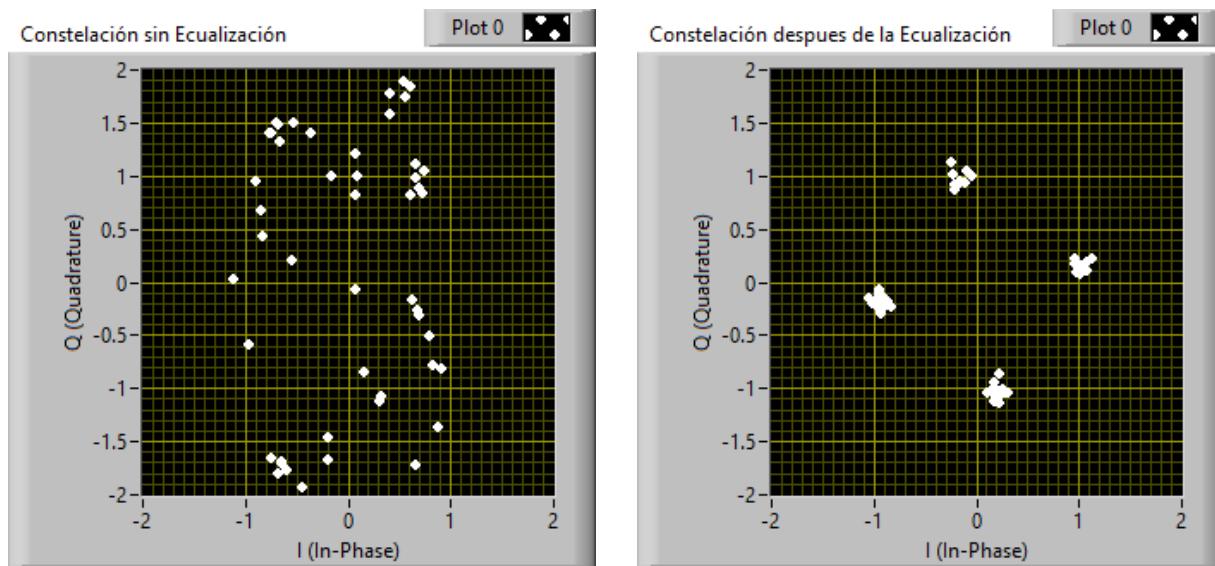


Figura 7.7 .- Comparación de la constelación QPSK sin ecualización (izquierda) y con ecualización (derecha)

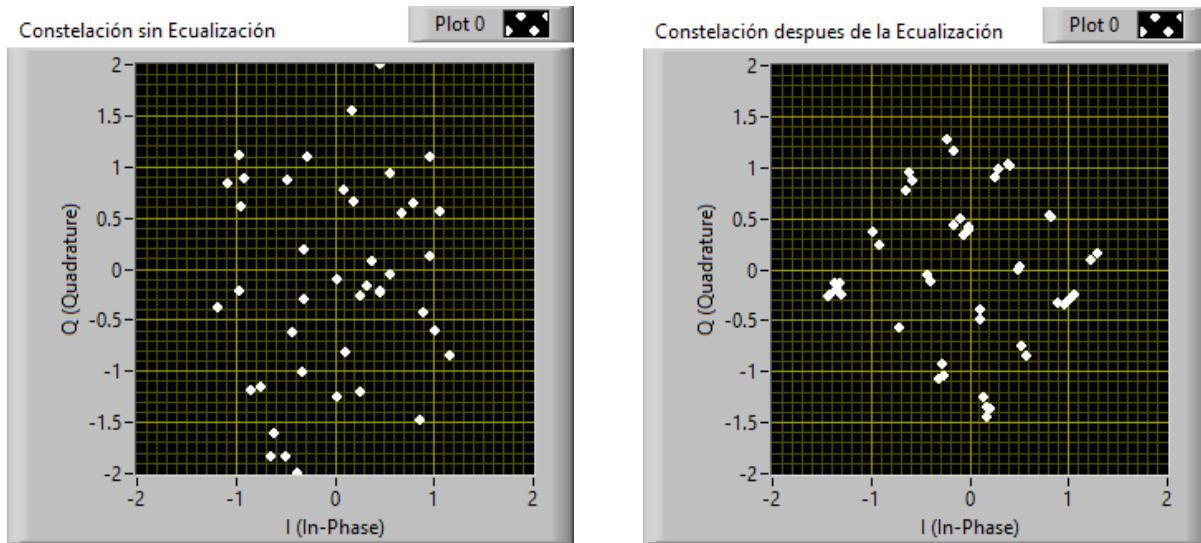


Figura 7.8 .- Comparación de la constelación 16-QAM sin ecualización (izquierda) y con ecualización (derecha)

7.1.4 Corrección de fase

De las Figuras 7.6, 7.7 y 7.8 de la sección anterior, es evidente que a pesar de que las constelaciones obtenidas después de la ecualización están muy bien definidas, dichas constelaciones se encuentran rotadas.

Por lo tanto, del ajuste realizado mediante las secuencias de entrenamiento en la etapa de estimación de canal y ecualización, el bloque de corrección de fase permite realizar un ajuste más preciso, usando las cuatro portadoras piloto insertadas en cada símbolo OFDM, con el objetivo de alinear las constelaciones de los datos modulados con modulación BPSK, QPSK, 16-QAM y 64-QAM y así lograr una mejor demodulación y decodificación de la información.

Las Figuras 7.9, 7.10, 7.11 y 7.12 presentan una comparación entre las constelaciones de entrada al bloque de corrección de fase (provenientes del bloque de ecualización) y las constelaciones obtenidas una vez que se ha aplicado la corrección en fase de acuerdo a la ecuación 6-11 del capítulo anterior, para modulaciones BPSK, QPSK, 16-QAM y 64-QAM.

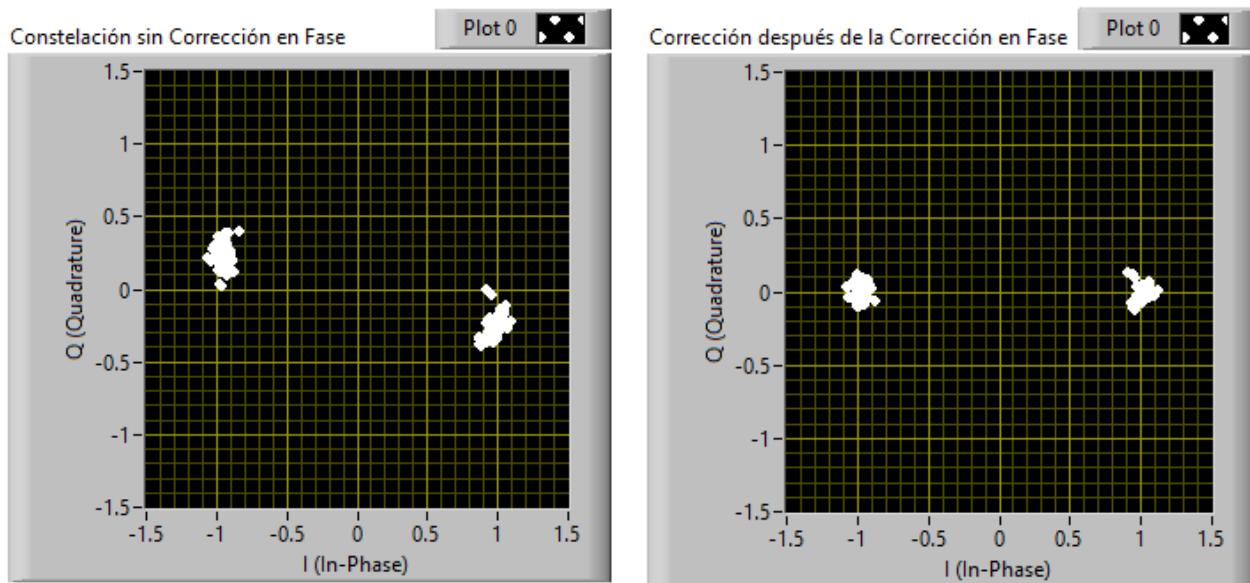


Figura 7.9 .- Comparación de la constelación BPSK sin corrección en fase (izquierda) y después de la corrección en fase (derecha)

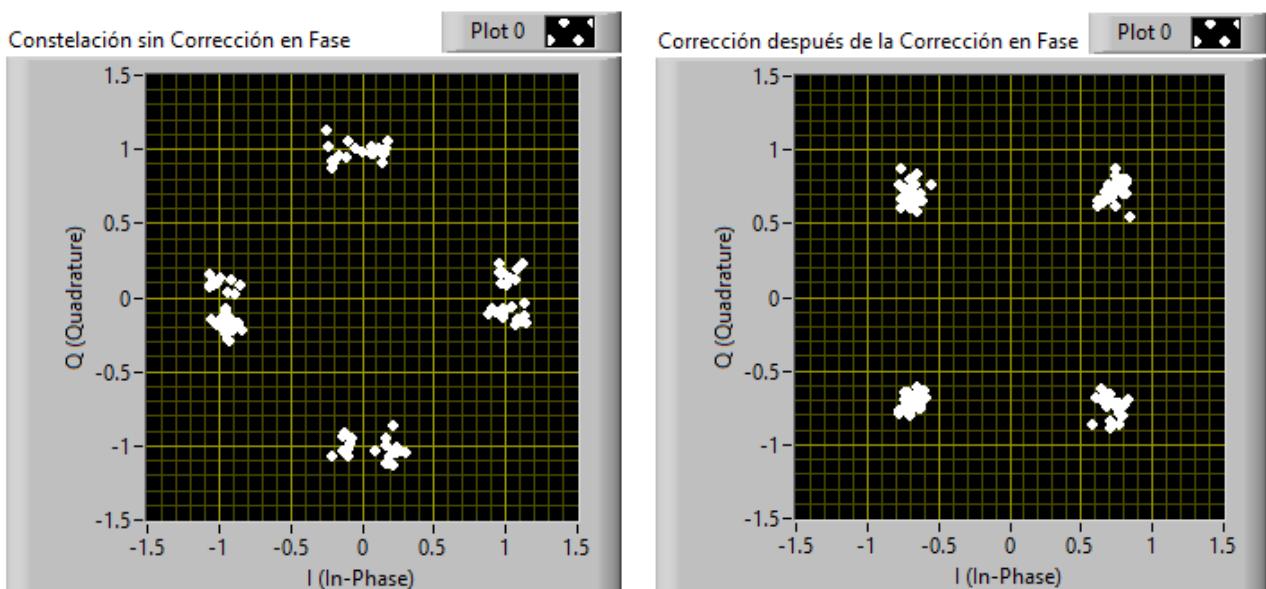


Figura 7.10 .- Comparación de la constelación QPSK sin corrección en fase (izquierda) y después de la corrección en fase (derecha)

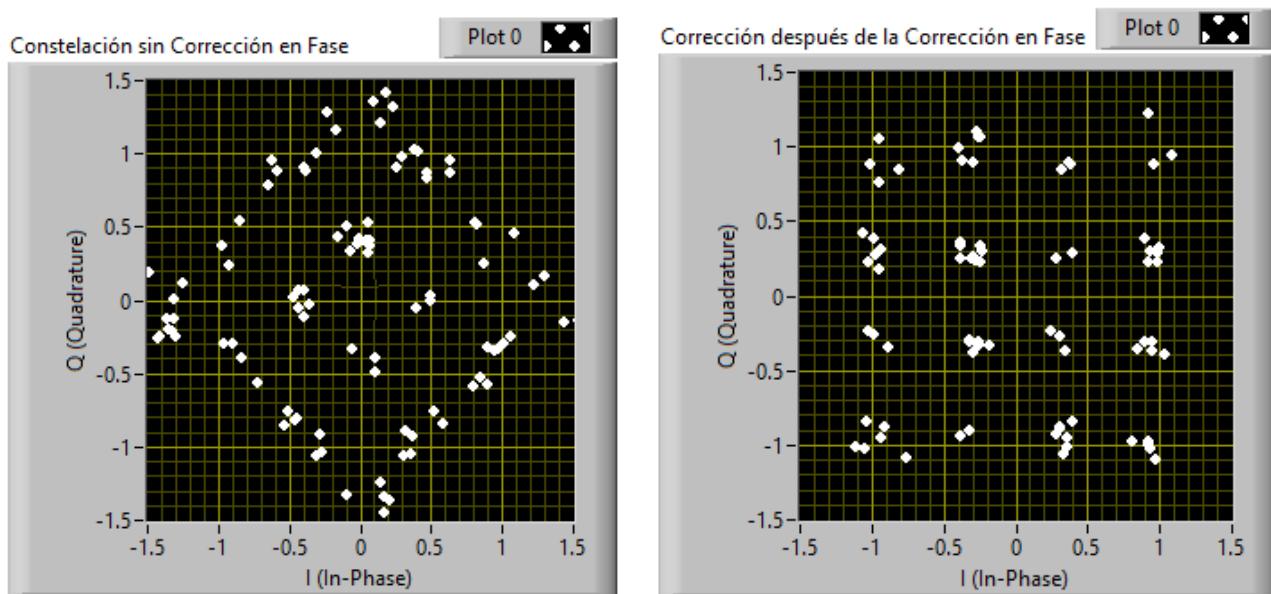


Figura 7.11 .- Comparación de la constelación 16-QAM sin corrección en fase (izquierda) y después de la corrección en fase (derecha)

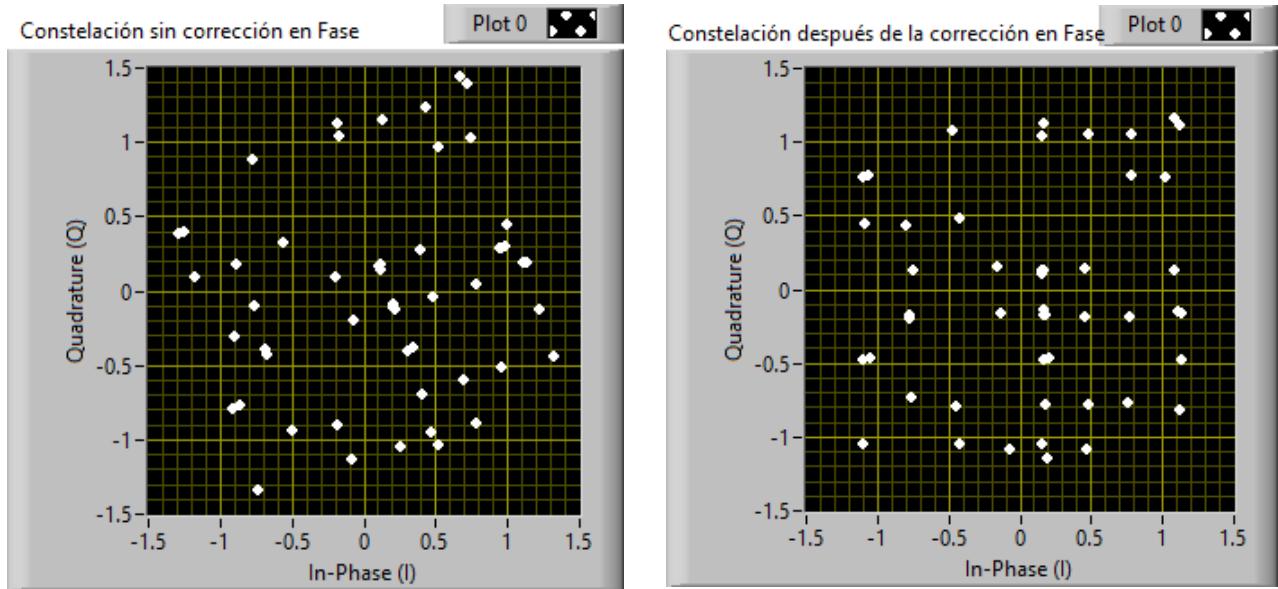


Figura 7.12 .- Comparación de la constelación 64-QAM sin corrección en fase (izquierda) y después de la corrección en fase (derecha)

7.1.5 Demodulación

Un ejemplo de la salida obtenida de la etapa de demodulación implementada en LabVIEW y la cual permite el mapeo de símbolos a bits se muestra en la Figura 7.13 para el caso de la modulación QPSK, adicionalmente se muestra el mapa de la constelación para éste tipo de modulación.

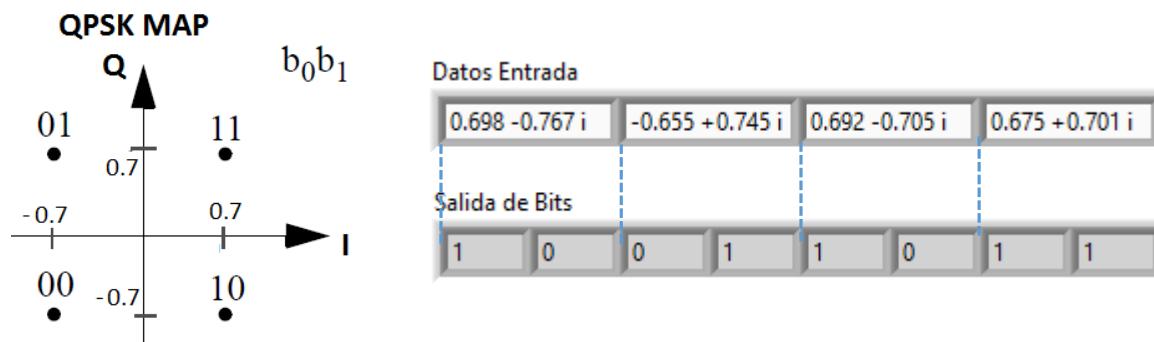


Figura 7.13 .- Prueba de la etapa de demodulación

7.2 Pruebas del enlace inalámbrico con un Tx y Rx OFDM

7.2.1 Especificaciones del sistema

Como parte final de este proyecto de investigación se estableció un enlace inalámbrico entre dos equipos USRP N210, cada uno operado por medio de una computadora, las cuales incluyen la capa física del estándar IEEE 802.11g respectivamente para la transmisión y para la recepción.

Las características y parámetros del sistema de transmisión se muestran en la Tabla 7.1, mientras que las características y parámetros que conforman el sistema de recepción se muestran en la Tabla 7.2.



Características del Transmisor OFDM		
Equipo	Componente	Tipo
USRP	Modelo	USRP N210
	Tarjeta Hija	SBX
Laptop	Modelo	Lenovo Y400
	Procesador	Intel Core i7 - 3630QM
	Disco Duro	1 TB
	Memoria RAM	8 GB
	NIC	Qualcomm Atheros AR8161 Gigabit Ethernet
	Sistema Operativo	Windows 8.1, 64 bits
	Software	LabVIEW 2013

Tabla 7.1 .- Parámetros del sistema de transmisión

Características del Receptor OFDM		
Equipo	Componente	Tipo
USRP	Modelo	USRP N210
	Tarjeta Hija	SBX
Laptop	Modelo	Lenovo Y50
	Procesador	Intel Core i7 - 4720HQ
	Disco Duro	1 TB
	Memoria RAM	8 GB
	NIC	Realtek RTL8111 Gigabit Ethernet
	Sistema Operativo	Windows 8.1, 64 bits
	Software	LabVIEW 2013

Tabla 7.2 .- Parámetros del sistema de recepción



Como se puede observar en las Tablas 7.1 y 7.2, ambos sistemas usan la tarjeta hija SBX la cual tiene un rango de operación de entre 400 – 4400 MHz. La tarjeta hija SBX fue seleccionada puesto que para las pruebas presentadas en éste capítulo se usó como frecuencia portadora **2.6 GHz**, éste valor de frecuencia central se eligió debido a su cercanía con la banda de los 2.4 GHz y la cual, por ser de uso libre, se encuentra saturada por otros sistemas de comunicación. La ventaja que se obtuvo al trabajar con una frecuencia de 2.6 GHz fue que en los ambientes en donde se realizaron las pruebas, esta frecuencia resultó libre de interferencias.

Otra característica importante del sistema de comunicación aquí presentado es el ancho de banda del canal, el cual fue de **2 MHz**. El valor del ancho de banda (frecuencia de muestreo) se eligió debido a que por la gran cantidad de procesamiento que se realiza en tiempo real en el equipo receptor, se observó que al aumentar el ancho de banda usado por el símbolo OFDM, la computadora no podía mantener la recepción activa y el software LabVIEW enviaba un error que solicitaba disminuir dicho ancho de banda. El efecto de disminuir el ancho de banda de 20 MHz a 2 MHz es el aumento de la duración de cada símbolo OFDM (de 4 us a 40 us) y de igual forma el aumento de la duración de cada secuencia de entrenamiento (de 8 us a 80 us), lo cual implica una reducción en la tasa de transmisión.

Para los escenarios aquí presentados se usaron dos tipos de antena, una antena omnidireccional modelo VERT 2450 (Figura 7.13) cuya frecuencia de operación es de 2.4 – 2.48 GHz y de 4.9 a 5.9 GHz, con una ganancia de 3 dBi. Por otro lado se usó una antena direccional logarítmica periódica LP0965 (Figura 7.14) implementada en una placa de circuito impreso o PCB (Printed Circuit Board), la cual trabaja en un rango de frecuencia entre 850 MHz y 6.5 GHz con una ganancia de 6 dBi.



Figura 7.14 .- Antena VERT2450

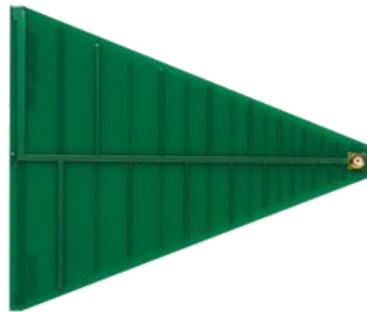


Figura 7.15 .- Antena LP0965

7.2.2 Escenario 1

La comunicación inalámbrica para éste escenario se llevó a cabo en un lugar cerrado (indoors), específicamente en la mesa de trabajo de un laboratorio de la Universidad de San José, California, en donde se realizó una estancia de investigación como parte de éste trabajo de tesis, para la implementación de los algoritmos de estimación de canal y ecualización. La Figura 7.16, muestra el escenario, el objetivo de esta prueba fue medir el número de bits en error (BER), en función de la relación señal a ruido (SNR) medida en el receptor.



Figura 7.16 .- Primer escenario para la prueba del enlace inalámbrico OFDM



Como lo muestra la Figura 7.16, el tipo de antenas utilizadas en este escenario fueron las antenas direccionales LP0965. La distancia entre las antenas para este escenario fue de 1 m de separación. Para lograr una variación en la relación señal a ruido se varió la ganancia del transmisor.

Para cada valor de SNR distinto se obtuvo la medición del BER mediante un programa en MATLAB, el cual se muestra en el Anexo G. El programa implementado realiza una comparación bit a bit entre los frames transmitidos y los frames recibidos llevando un control del número de secuencia de cada frame para finalmente arrojar la relación entre el número de bits en error y el número total de bits (BER).

Los mensajes enviados por el transmisor fueron de una longitud de 100 Bytes (800 bits) en total. El contenido del campo de datos de las tramas enviadas (72 bytes) se muestra en la Figura 7.17, dicho mensaje se encuentra propuesto en el anexo G del estándar IEEE 802.11. Al campo de Datos se le agregó el encabezado correspondiente a la capa MAC para una trama de datos, tal como direcciones MAC, CRC, bits de relleno, etc.

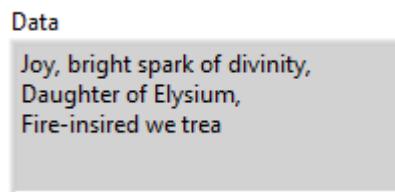


Figura 7.17 .- Contenido del campo de datos de las tramas enviadas

La tasa de transmisión (Data Rate) del sistema está dada por la ecuación 7-1:

$$(7-1) \quad \text{Data Rate} = \frac{1}{T_s} * cc * m * N_{FFTdata}$$

Donde:

cc es la tasa de codificación

m es el número de bits por símbolo

N_{FFT}data es el número de portadoras de datos

T_s es el tiempo del símbolo OFDM y esta dado por las ecuaciones 7 – 2 y 7 – 3



$$(7-2) \quad \Delta f = \frac{BW}{N_{FFT}} = \frac{2 \text{ MHz}}{64} = 31250 \text{ Hz}$$

$$(7-3) \quad Ts = 1.25 \left(\frac{1}{\Delta f} \right) = 4 \times 10^{-5} \text{ s}$$

Donde:

Δf es la separación entre portadoras

BW es el ancho de banda

N_{FFT} es el número total de portadoras

Entonces para los esquemas de modulación BPSK 1/2 (Ecuación 7-4) y 16-QAM 1/2 (Ecuación 7-5) se tiene:

$$(7-4) \quad \text{Data Rate}_{BPSK \frac{1}{2}} = \frac{1}{4 \times 10^{-5}} * \frac{1}{2} * 1 * 48 = 600 \text{ Kbps}$$

$$(7-5) \quad \text{Data Rate}_{QPSK \frac{1}{2}} = \frac{1}{4 \times 10^{-5}} * \frac{1}{2} * 4 * 48 = 2.4 \text{ Mbps}$$

En esta prueba se observó una mejora en el espectro de la señal transmitida cuando la salida del transmisor se multiplicaba por un factor, encontrándose que el mejor valor del multiplicador para BPSK fue de 0.5 y para 16-QAM fue de 0.125.

Los resultados de BER vs SNR para la modulación BPSK 1/2 se muestran en la tabla 7.3 y su respectiva gráfica se muestra en la Figura 7.18.

BPSK 1/2, AMP=0.5	
SNR	BER
12.2	9.96E-04
13.3	9.96E-04
16	6.60E-04
17.8	3.30E-04
19.6	2.40E-04
21.8	0.00E+00

Tabla 7.3 .- SNR vs BER para el esquema de modulación BPSK 1/2

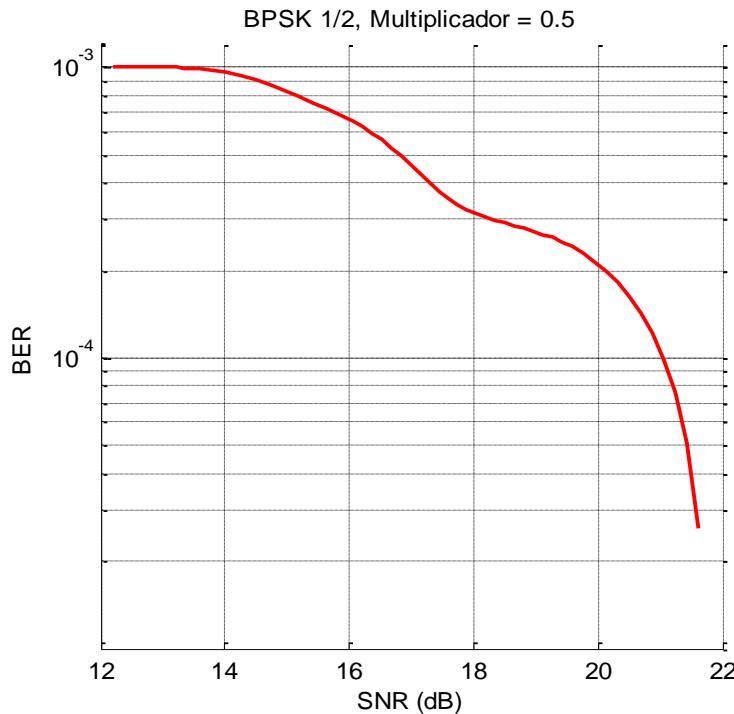


Figura 7.18 .- Gráfica de BER vs SNR para modulación BPSK 1/2

Para el esquema de modulación 16-QAM 1/2, los resultados de BER vs SNR se muestran en la Tabla 7.4 y la gráfica se muestra en la Figura 7.19.

16-QAM, AMP=.125	
SNR	BER
12.6	0.586
13.5	0.214
15.6	0.0024
17.9	0.0015
18.9	4.99E-04
19.2	4.99E-04
22.7	0

Tabla 7.4 .- SNR vs BER para el esquema de modulación 16-QAM 1/2

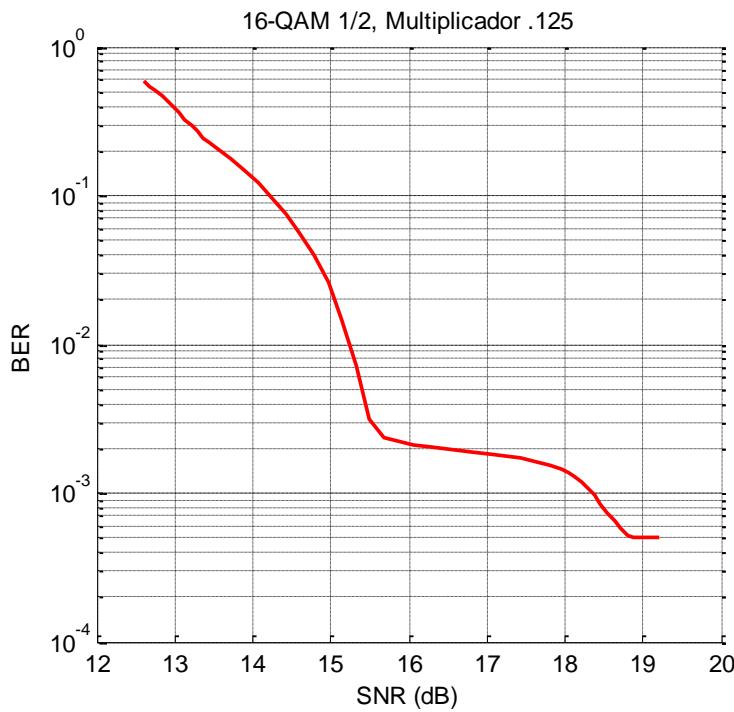


Figura 7.19.- Gráfica de BER vs SNR para modulación 16 QAM 1/2

Como se puede observar en las Figuras 7.18 y 7.19, la modulación BPSK presenta un comportamiento más robusto, puesto que el BER es menor en cualquier punto de SNR dado, sin embargo, la modulación 16-QAM brinda una mayor tasa de transferencia como se demostró con las ecuaciones 7-4 y 7-5.

7.2.3 Escenario 2

La comunicación entre transmisor y receptor para este escenario se llevó a cabo en un ambiente de interiores, el objetivo fue determinar el comportamiento del BER vs la distancia entre equipo transmisor y receptor. Mientras que el equipo receptor se fijó en una posición, se varió la posición del equipo transmisor.

El escenario en donde se realizó la medición se muestra en la Figura 7.20. Se trata de un pasillo del edificio Q “Luis G. Valdés Vallejo” de la Universidad Nacional Autónoma de México. El pasillo es de aproximadamente 1.5 m de ancho. Cada equipo USRP se ubicó a una altura de 1m, para este escenario se usaron las antenas de tipo direccional LP0965 y las antenas VERT 2450.

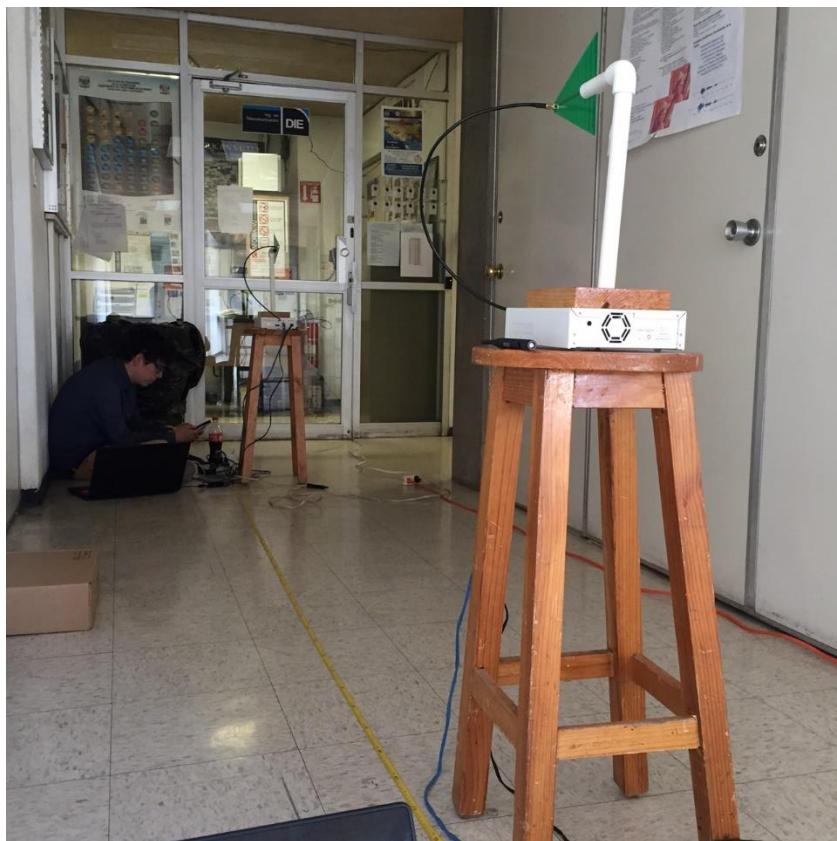


Figura 7.20.- Segundo escenario para la prueba del enlace inalámbrico OFDM

Los resultados obtenidos con las antenas LP0965 para las modulaciones BPSK 1/2 y 16-QAM 1/2 se muestran a continuación en la Tabla 7.6 y su respectiva gráfica en la Figura 7.21.

Distancia [m]	BER, BPSK 1/2	BER, 16-QAM 1/2
1	0	2.49E-04
2	0	9.995E-04
3	0.0025	3.500E-03
4	0.0024	0.0045
5	0.003	0.0035
6	0.0045	0.0035
7	0.006	0.0055
8	0.003	0.00499
9	0.026	0.042
10	0.0375	0.1317
11	0.1562	0.1664

Tabla 7.6 .- Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 2 con antena LP0965

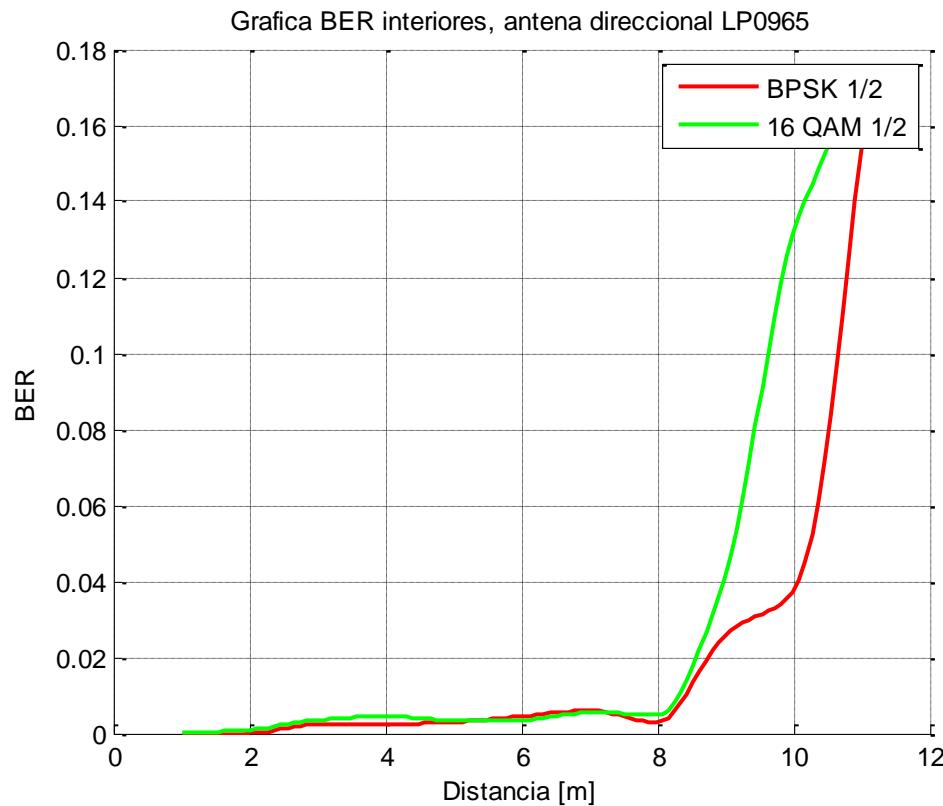


Figura 7.21.- Distancia vs BER para el segundo escenario con modulaciones BPSK 1/2 y 16QAM 1/2, antena LP0965

Posteriormente se realizó la misma prueba pero trabajando con las antenas VERT2450, de igual forma se obtuvo el BER contra distancia para las modulaciones BPSK 1/2 y 16-QAM 1/2 y los resultados se muestran en la Tabla 7.7 y su respectiva gráfica en la Figura 7.22.

Distancia [m]	BER, BPSK 1/2	BER, 16-QAM 1/2
2	0.3178	0.3555
4	0.1562	0.2901
6	0.1813	0.5609
8	0.3503	0.9531

Tabla 7.7 .- Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 2 con antena VERT 2450

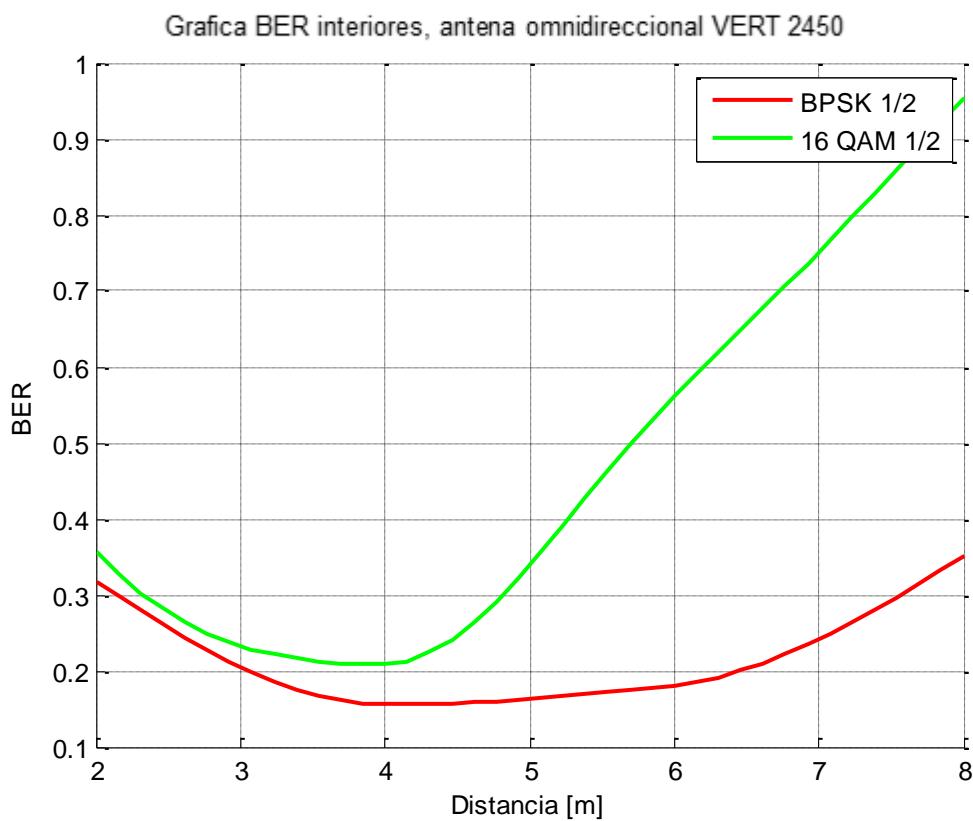


Figura 7.22.- Distancia vs BER para el segundo escenario con modulaciones BPSK 1/2 y 16QAM 1/2, antena VERT 2450

De la Figura 7.21 (Antena LP0965) se puede apreciar que ambas modulaciones tienen un comportamiento esperado, puesto que en general la modulación BPSK 1/2 contiene un número menor de bits en error en comparación con el esquema 16-QAM 1/2 llegando incluso a valores de hasta 0 bits en error como es el caso de las distancias a 1 y 2 m, en cambio el mejor comportamiento para 16 QAM es cuando se pierden 3 bits de 10000 bits enviados. El máximo alcance del enlace inalámbrico fue de 11 m aunque a esa distancia se perdió aproximadamente un 15% de la información.

Como se puede observar en la Figura 7.22, la antena VERT 2450 tiene un comportamiento deficiente para ambos casos de modulación, llegando a perder en el mejor de los casos el 15% y 29% de la información transmitida para BPSK y 16-QAM respectivamente. Cabe destacar que el comportamiento deficiente de la antena VERT 2450, se puede atribuir a que se usó 150 MHz por encima de su frecuencia de operación.

7.2.4 Escenario 3

Finalmente la última prueba realizada con el sistema de comunicación inalámbrica se realizó en un ambiente de exteriores, como lo muestra la Figura 7.23



Figura 7.23.- Tercer escenario para la prueba del enlace inalámbrico OFDM

De igual forma que en el escenario 2, la estación receptora permaneció fija mientras que la estación transmisora varió su posición para lograr obtener el BER en función de la distancia entre ambos equipos.

Los resultados obtenidos para las modulaciones BPSK 1/2 y 16 QAM 1/2, se muestran en la Tabla 7.8 y su gráfica correspondiente se aprecia en la Figura 7.24.

Distancia [m]	BER, BPSK 1/2	BER, 16-QAM 1/2
2	0.0015	0.0015
9	0.003	0.012
16	0.009	0.016
21	0.1246	0.1504

Tabla 7.8 .- Distancia vs BER para BPSK 1/2 y 16-QAM 1/2 en el escenario 3

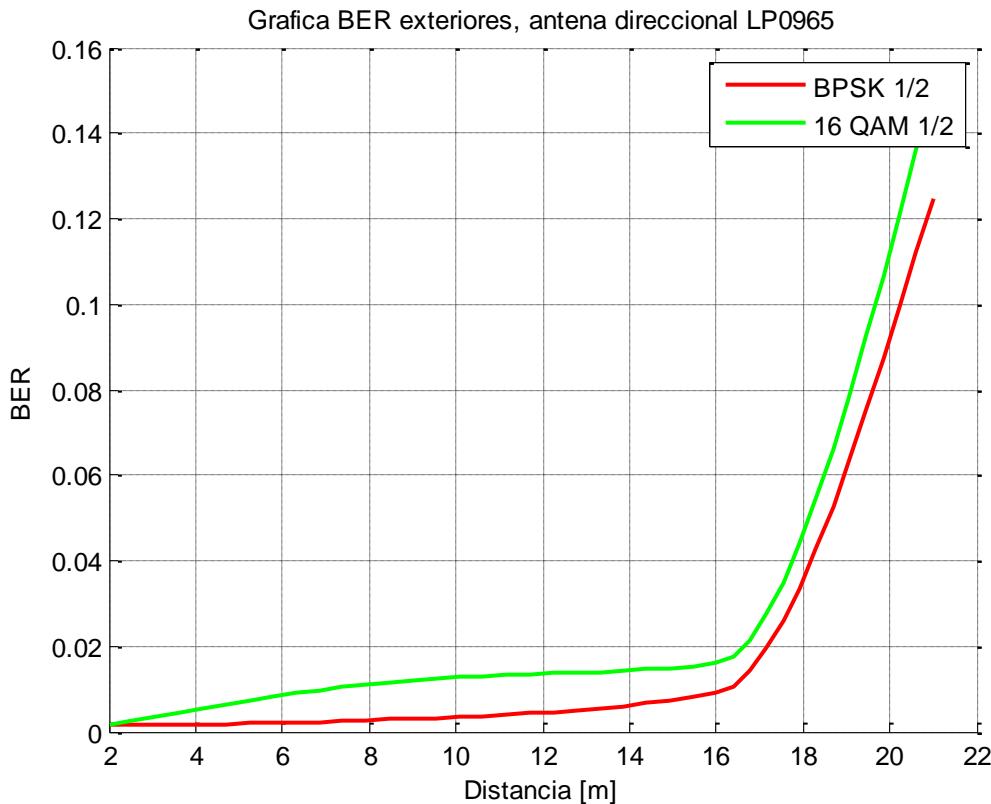


Figura 7.24.- Distancia vs BER para el tercer escenario con modulaciones BPSK 1/2 y 16QAM 1/2, antena LP0965



Finalmente con los resultados obtenidos en éste escenario, se puede observar que la ausencia de obstáculos en un ambiente de exteriores permite que la distancia del enlace sea mucho mayor en comparación con el enlace de interiores. La distancia máxima alcanzada con este escenario fue de 21m perdiendo entre el 12 y el 15 % de bits enviados. De igual forma con la ayuda de este escenario se demostró la mayor robustez de la modulación BPSK 1/2 sobre el esquema de modulación 16-QAM 1/2.



CONCLUSIONES

En el presente trabajo de tesis se implementaron de manera práctica y se evaluaron los algoritmos de capa física que conforman un receptor OFDM basado en el estándar IEEE 802.11. Se analizó el funcionamiento y los principios matemáticos que rigen cada uno de los bloques que conforman éste proyecto.

Al término de éste trabajo de tesis se puede destacar el gran potencial de los radios definidos por software, se pudo comprobar que la plataforma SDR ofrece la oportunidad de definir un sistema de comunicaciones inalámbrico cuyas funciones más importantes como el tipo de modulación, la frecuencia central, el ancho de banda y todas las funciones de la capa física de un estándar se pueden implementar y evaluar sin ningún cambio en hardware.

Con el estudio del estado arte de los prototipos diseñados e implementados en una plataforma SDR hasta la fecha, se motivó a realizar el primer receptor en LabVIEW para OFDM trabajando con SDR que soporta los esquemas de modulación BPSK, QPSK, 16-QAM y 64-QAM del que se tiene conocimiento. Además con las funciones de capa física de interleaver, scrambler y codificación convolucional.

De los resultados obtenidos, es importante destacar las limitaciones que se encontraron al realizar la implementación del proyecto, en particular el máximo ancho de banda alcanzado, el cual fue de 2 MHz en comparación de los 20 MHz definidos en el estándar IEEE 802.11g. Esta reducción puede ser debida al buffer dedicado al software LabVIEW para la entrada de datos proveniente de la tarjeta de red, además del exceso de procesamiento que la PC debe realizar en tiempo real para la decodificación de cada uno de los frames OFDM transmitidos.



La reducción en el ancho de banda repercutió directamente con la tasa de transmisión, la cual se redujo en un factor de 10, obteniéndose velocidades de 600 Kbps y 2.4 Mbps para los esquemas de modulación BPSK 1/2 y QPSK 1/2 respectivamente.

Por otro lado, se presentó el comportamiento del sistema en 3 escenarios distintos, 2 en interiores y 1 en exteriores y se realizó un análisis comparativo entre los esquemas de modulación BPSK 1/2 y 16 QAM 1/2, para los 3 casos se obtuvo, como era de esperarse un mejor comportamiento en términos de menor BER para el caso de la modulación BPSK.

En el primer escenario en interiores, se puede destacar que el sistema con modulación BPSK 1/2 con un SNR de 12 dB perdió solamente 1 de cada 10000 bits, mientras que para la modulación 16-QAM no fue soportado con ese nivel de SNR pues se perdieron casi la mitad de los paquetes, reduciéndose drásticamente el valor del BER (para 16 QAM) con un valor de SNR de 19 dB.

Del segundo escenario, se puede concluir que la máxima distancia soportada es de 8m para ambas modulaciones, puesto que después de ésta distancia el BER crece de manera exponencial. Cabe mencionar que esto se observó para el tipo de antenas LP0965, mientras que para las antenas omnidireccionales VERT 2450 su comportamiento no fue satisfactorio, debido a que se usó en un rango de frecuencia distinto al especificado por el fabricante, 150 MHz arriba de su frecuencia de operación (para fines puramente experimentales).

Finalmente del tercer escenario se pudo observar los beneficios de un ambiente de exteriores libre de obstáculos, pues el alcance del sistema de comunicaciones se incrementó a más del doble de la distancia máxima de interiores, lográndose un alcance máximo de hasta 21m con aproximadamente el 15% de los bits perdidos y de 16m con aproximadamente 1% de los bits perdidos.



En general, se puede concluir que actualmente la plataforma SDR y en específico el USRP N210 tiene ciertas limitaciones para ser implementado en ambientes reales, puesto que los APs tradicionales (para el caso de redes inalámbricas de área local) tienen un comportamiento superior, en cuanto a velocidad y BER (10^{-6}), sin embargo, la tecnología SDR permite implementar y evaluar diversas tecnologías de capa física sin necesidad de cambiar el equipo usado, simplemente seleccionando el software apropiado.

Debido a que la tecnología avanza a pasos agigantados, sin lugar a dudas, será posible la obtención de sistemas cada vez más veloces, tanto PCs, como plataformas SDR que permitan lidiar con los problemas de velocidad que aquí se encontraron. Entonces, al ser capaces de desarrollar un sistema de comunicaciones diseñado e implementado vía software se tienen las siguientes implicaciones:

- Se pueden desarrollar y evaluar con mayor facilidad los principios básicos que rigen las tecnologías de comunicaciones inalámbricas actuales y por tanto, proponer otras técnicas para lograr comunicaciones más eficientes en cuanto a velocidad y eficiencia (BER).
- Se pueden tener dispositivos que soporten múltiples tecnologías con el fin de garantizar una mayor interoperabilidad entre dichas tecnologías y de igual forma, asegurarse que los dispositivos estén siempre conectados, por ejemplo, si un tipo de tecnología no se encuentra disponible en un cierto lugar, el dispositivo puede cambiar de tecnología mediante otro módulo de software y operar normalmente.



TRABAJO FUTURO

Definitivamente el desarrollo de éste trabajo proporciona un amplio panorama en cuanto al diseño e implementación de un receptor OFDM usando la plataforma SDR, por otro lado, queda claro que aún hay mucho trabajo por hacer, es indispensable plantear nuevas líneas de desarrollo para complementar éste proyecto y por lo tanto se propone:

- Realizar la implementación del proceso de transmisión / recepción OFDM en un solo dispositivo SDR.
- Realizar la implementación en software de la capa MAC del estándar IEEE 802.11 para la plataforma SDR, con el objetivo de lograr un sistema de comunicación inalámbrico completo que incluya todas las funciones descritas en el estándar, tanto las de capa PHY, las cuales fueron implementadas en éste trabajo de tesis, como las funciones de capa MAC revisadas en el capítulo 3, como son: el control de acceso al medio (CSMA/CA), el algoritmo de exponential back-off y el mecanismo RTS/CTS.
- Explorar otros equipos en la plataforma SDR que permitan mejorar la recepción de los símbolos OFDM, en cuanto al máximo ancho de banda permitido, como es el caso del USRP E310, el cual tiene un procesador integrado que permite al SDR operar de forma autónoma sin necesidad de una PC que se encargue del procesamiento en banda base. Adicionalmente se puede explorar la posibilidad de reconfigurar el FPGA para que sea la unidad encargada de algoritmos que requieren una gran cantidad de procesamiento, como es el caso de la detección de los símbolos OFDM.



REFERENCIAS

- [1] J. Geier, "Wireless LANs", Sams, Second Edition, 2002.
- [2] IEEE Standard 802.11-1997 for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.
- [3] IEEE Standard 802.11a-1999, High Speed Physical Layer for the 5 GHz band.
- [4] IEEE Standard 802.11b-1999, Higher Speed Physical Layer (PHY) Extension in the 2.4 GHz band.
- [5] IEEE Standard 802.11g-2003, Further Higher Data Rate Extension in the 2.4 GHz.
- [6] IEEE Standard 802.11n-2009, Enhancements for Higher Throughput.
- [7] IEEE Standard 802.11ac-2013, Enhancements for very High Throughput for operation in bands below 6 GHz.
- [8] IEEE Standard 802.11ad-2012, Enhancements for very High Throughput in the 60 GHz band.
- [9] J. Mitola, "Software Radios Survey, Critical Evaluation and Future Directions", National Telesystems Conference, IEEE, 1992.
- [10] A. Di Stefano, G. Fiscelli, C.G. Giacconi, "An FPGA-Based Software Defined Radio Platform for the 2.4 GHz ISM Band", Research in Microelectronics and Electronics, IEEE, 2006.
- [11] Y. Tachwali, H. Refai, "Implementation of a BPSK Transceiver on Hybrid Software Defined Radio Platforms", 3rd International Conference on Information and Communication Technologies: from Theory to Applications, IEEE, 2008.



- [12] A. Marwanto; M. A. Sarijari, N. Fisal, S. Yusof, A. Rashid, "Experimental Study of OFDM Implementation Utilizing GNU Radio and USRP-SDR", 9th Malaysia International Conference on Communications, IEEE, 2009.
- [13] J. R. Gutiérrez-Agullo, B. Coll-Perales and J. Gozalvez, "An IEEE 802.11 MAC Software Defined Radio Implementation for Experimental Wireless Communications and Networking Research", Wireless Days, IEEE, 2010.
- [14] P. Fuxjager, A. Constantini, D. Valerio, P. Castiglione, G. Zacheo, T. Zemen, F. Ricciato, "IEEE 802.11p Transmission Using GNURadio", 6th Karlsruhe Workshop on Software Radios (WSR), 2010.
- [15] B. Bloessl, M. Segata, C. Sommer and F. Dressler, "An IEEE 802.11a/g/p OFDM Receiver for GNU Radio", ACM SIGCOMM 2013, 2nd ACM SIGCOMM Workshop of Software Radio Implementation Forum, 2013.
- [16] N. Shahin, N. J. LaSorte, S. A. Rajab, H. Refai, "802.11g Channel Characterization Utilizing LabView and NI-USRP", IEEE International Instrumentation and Measurement Technology Conference, IEEE, 2013.
- [17] Lin LEI; Chen SONG, Taige Zhang, "Performance evaluation for OFDM Link Based on LabView and USRP", 4TH International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), IEEE, 2014.
- [18] W. Stallings, "Wireless Communications and Networks", Pearson, Second Edition, 2005.
- [19] R. Prasad, "OFDM for Wireless Communications Systems", Artech House, 2004.
- [20] Y. Li, G. Stuber, "Orthogonal Frequency Division Multiplexing for Wireless Communications", Springer, 2006.
- [21] Whitepaper "Software-Defined Radio: A Technology Overview", Wipro Technologies, 2002.



- [22] M. Srilatha, R. Hemalatha, T. Sri Aditya, P. Anantha, "Knowledge based Analysis of Software Defined Radio for Wireless Communication: A Preliminary Survey", International Journal of Computer Applications, IEEE, 2013.
- [23] M. Dardaillon, K. Market, T. Risset, A. Scherrer, "Software Defined Radio Architecture Survey for Cognitive Testbeds", Wireless Communications and Mobile Computing Conference, IEEE, 2012.
- [24] USRP: www.ettus.com
- [25] QuickSilver: www.srl-llc.com
- [26] SORA: <http://research.microsoft.com/en-us/projects/sora/>
- [27] M. Fähnle, "Software-Defined Radio with GNU Radio and USRP2 Hardware Frontend: Setup and FM/GSM Applications", Bachelor Thesis, Institute of Communication Technology, Germany, 2010.
- [28] USRP N200/210 Networked Series Datasheet:
http://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf
- [29] "LabVIEW Fundamentals", National Instruments, 2005.
- [30] E. Sourour, H. El-Ghoroury and D. McNeill, "Frequency Offset Estimation and Correction in the IEEE 802.11a WLAN", Vehicular Technology Conference, IEEE, 2004.
- [31] R. Morelos-Zaragoza, "Frequency Diversity for Frequency-selective multipath channels: OFDM", EE161: Digital Communication Systems, San Jose State University, 2012.
- [32] R. Walpole, H. M. Raymond, S. L. Myers, K. Ye, "Probabilidad y Estadística para Ingeniería y Ciencias", Pearson, Octava Edición, 2007.



GLOSARIO

A

Ancho de banda. Intervalo de frecuencias que son usadas para la transmisión de una señal.

Atenuación. Reducción o pérdida de energía de una señal.

B

Banda base. Conjunto de señales que no han pasado por el proceso de modulación

BER. Relación del número de bits en error entre el número total de bits enviados en un intervalo de tiempo dado.

Bluetooth. Estándar de comunicación inalámbrica de corto alcance para la interconexión de teléfonos celulares, computadoras y otros dispositivos.

bps. Bits por segundo.

C

Capa de enlace de datos. Es la capa 2 del modelo OSI, se encarga de dividir los datos en tramas más pequeñas, envío de confirmaciones y de asegurar

que las tramas recibidas estén libres de errores.

Capa física. Capa más baja del modelo OSI, define el hardware necesario, el cableado, la frecuencia y otras características respecto a la señal que contiene la información binaria.

Ciclo prefijo: Sección de una señal de datos que se ante pone a ésta para lidiar con efectos multirayectoria.

Constelación. Representación gráfica de una señal modulada, la cual está formada por dos ejes: fase y cuadratura.

D

Decodificación. Convertir una palabra codificada a su forma original.

Demodulación. Proceso de separar una señal en banda base de la señal portadora.

Detección de error. Proceso mediante el cual se valida que los bits enviados hayan llegado libres de errores en el receptor.

**E**

Ecualización. Se refiere al proceso de hacer que las portadoras que conforman un símbolo OFDM sean uniformes.

Estimación del canal. Proceso mediante el cual se obtiene la respuesta del medio inalámbrico a una cierta frecuencia (subportadora).

Estándar. Conjunto de especificaciones adoptadas en la industria que permite la compatibilidad entre distintos productos.

F

Fase. Ángulo inicial de una señal sinusoidal.

Frecuencia. Número de ciclos por segundo de una señal.

H

Hertz. Unidad de frecuencia.

I

IEEE. Instituto de Ingenieros Electricos y Electrónicos, es una asociación encargada de promover y desarrollar normas y especificaciones internacionales en el área de computación, ingeniería electrónica y eléctrica.

Interferencia. Perturbación de una señal de datos debida a otras fuentes de señales electromagnéticas no deseadas.

ISM. Bandas de radio frecuencias reservadas al uso industrial, científico y médico.

M

Modulación. Proceso de modificación de una señal periódica (señal portadora) por la señal que contiene la información a transmitir.

O

OFDM. Multiplexado por división de frecuencias ortogonales, es una técnica de modulación en donde la señal es enviada a través de distintas frecuencias, las cuales son ortogonales entre sí.

OSI. Modelo de referencia (pila de protocolos) que describe la comunicación entre dos elementos de red y está dividido en 7 capas.

P

Portadora. Señal periódica con cierta frecuencia (frecuencia central) apropiada para enviarse por el medio, en la cual se envía la información

Protocolo. Define las reglas y convenciones para la comunicación entre dos dispositivos.

Punto de Acceso. Dispositivo que permite a otros equipos acceder inalámbricamente a una red.

**Q**

QAM. Modulación por amplitud en cuadratura, es un método de modulación en donde se usan dos señales portadoras desfasadas 90°, llamadas I (In-Phase) y Q (Quadrature).

S

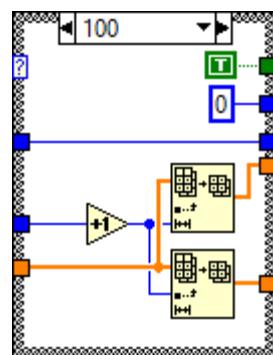
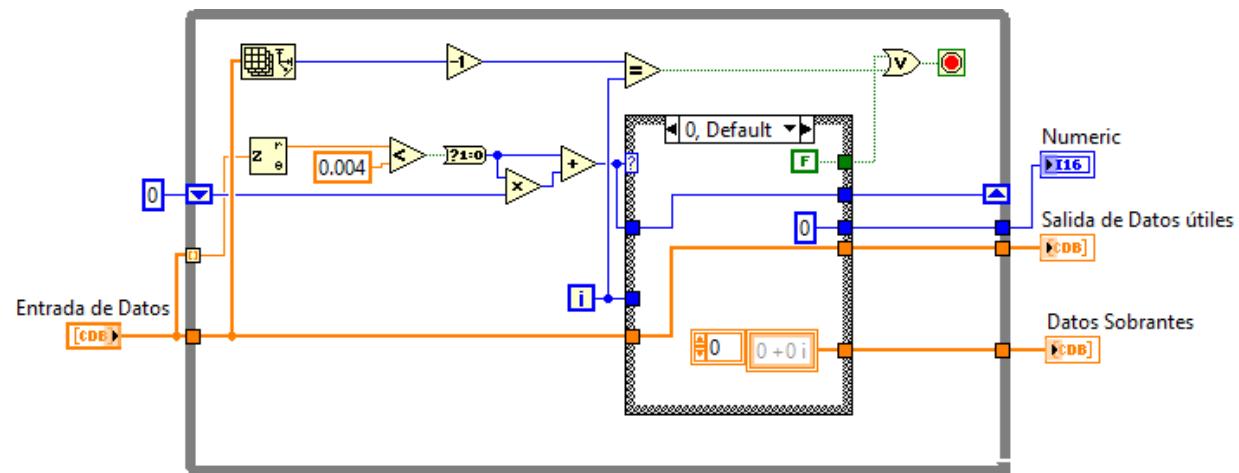
SDR. Radio definido por software, tipo de radio cuyas funciones y características son definidas vía software.

Señal. Onda electromagnética propagada a través de un medio de transmisión.

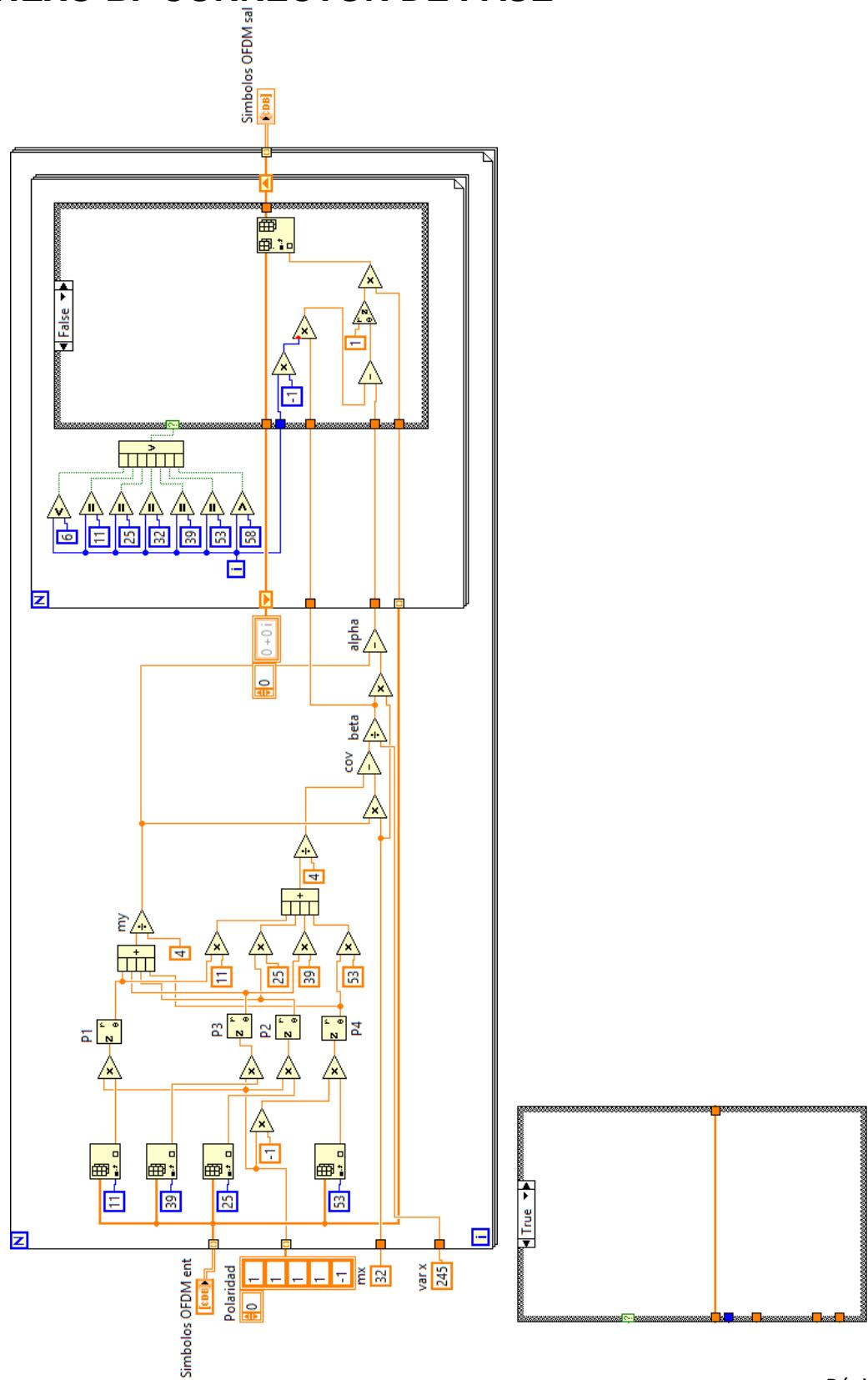
W

Wi-Fi. Es una marca registrada por la Wi-Fi Alliance, la cual se refiere a la certificación otorgada a los dispositivos con conexión inalámbrica que cumplen con el estándar IEEE 802.11

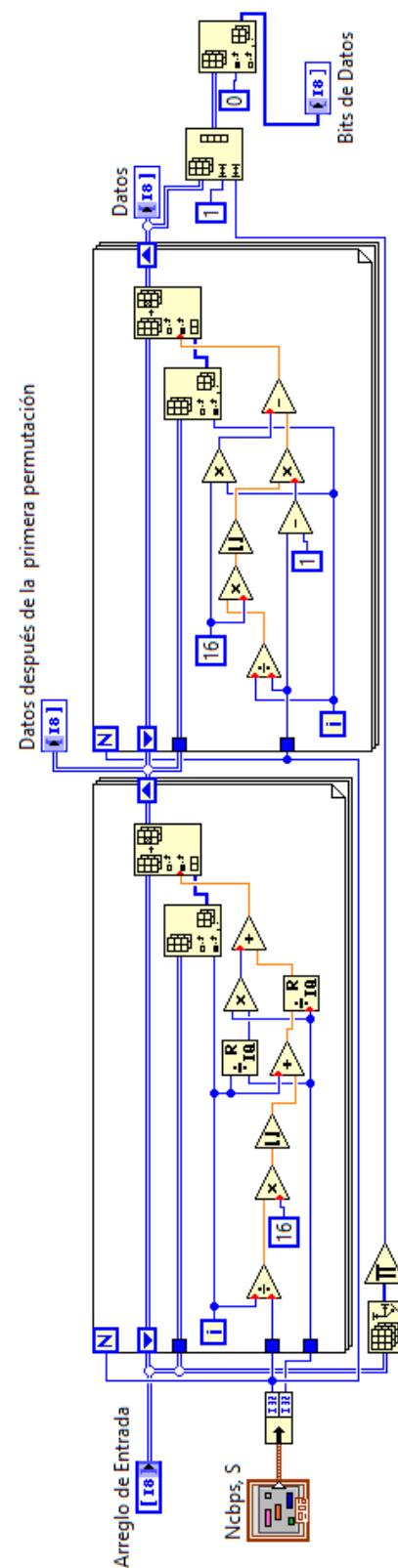
ANEXO A. SELECTOR DE DATOS



ANEXO B. CORRECTOR DE FASE



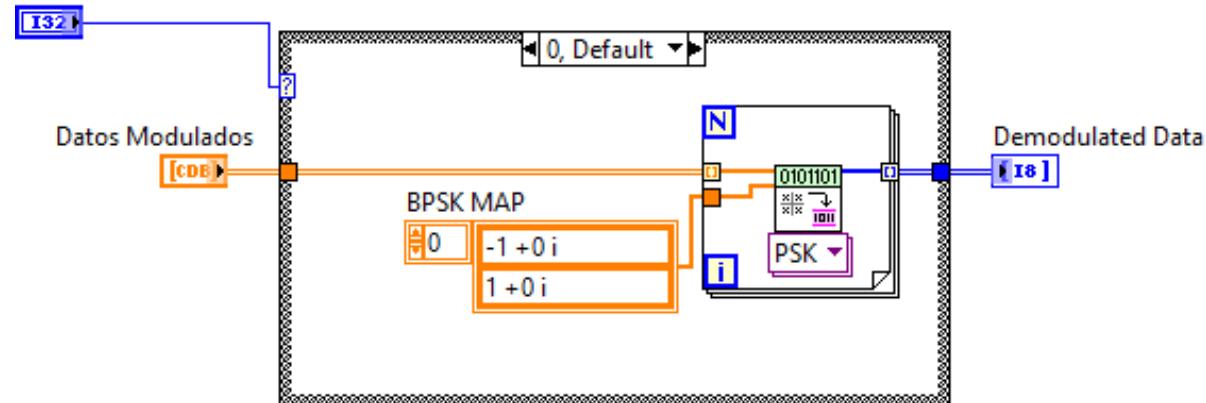
ANEXO C. DEINTERLEAVER



ANEXO D. DEMODULADOR

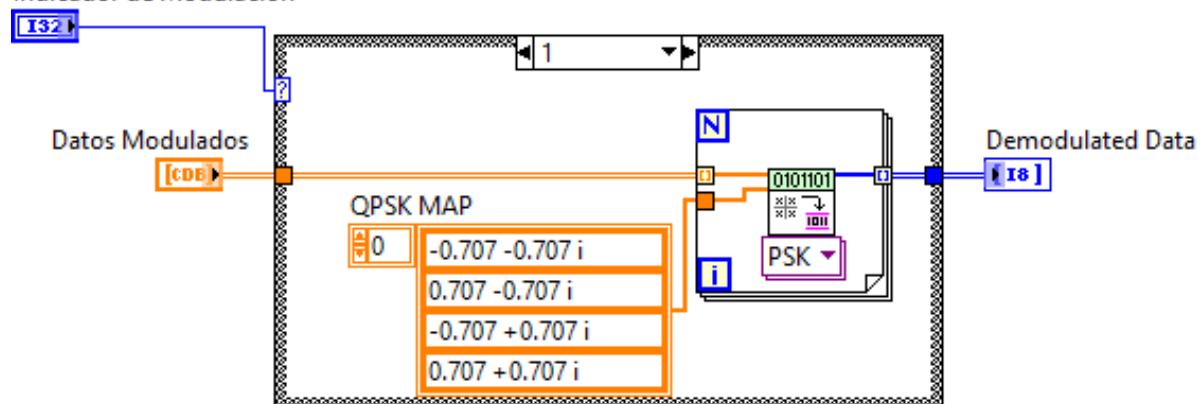
Caso 0: BPSK

Indicador de Modulación

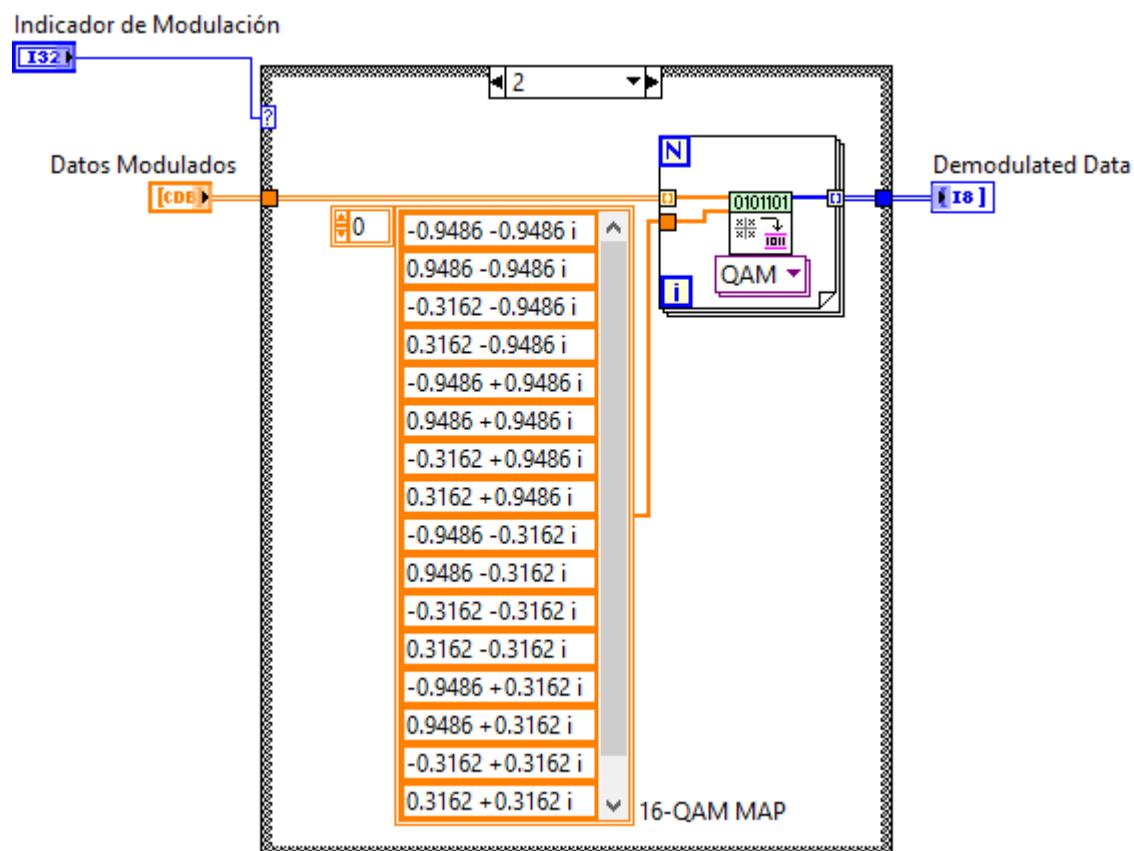


Caso 1: QPSK

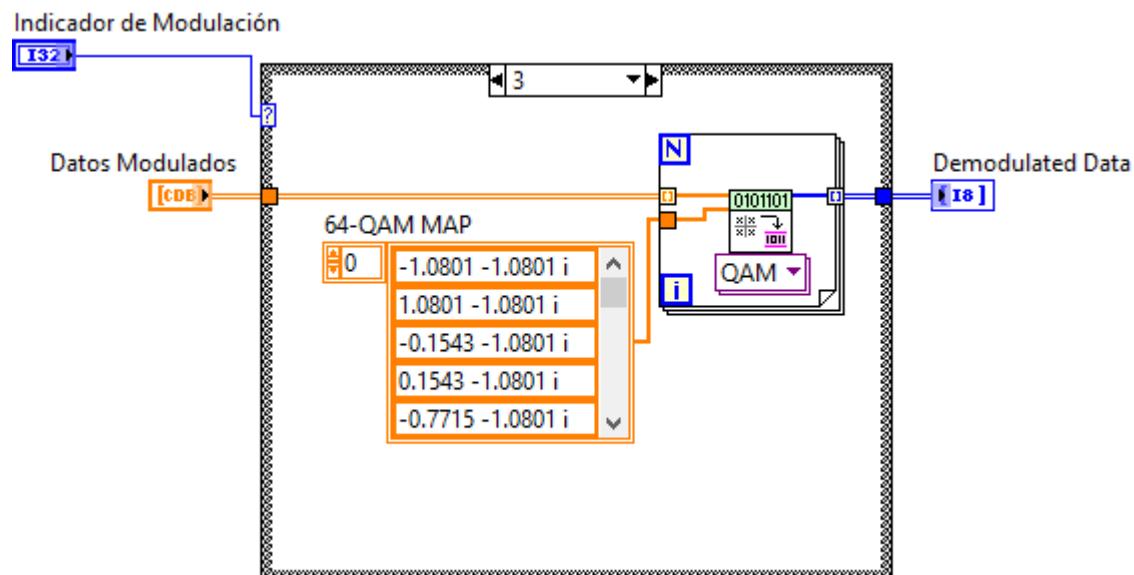
Indicador de Modulación



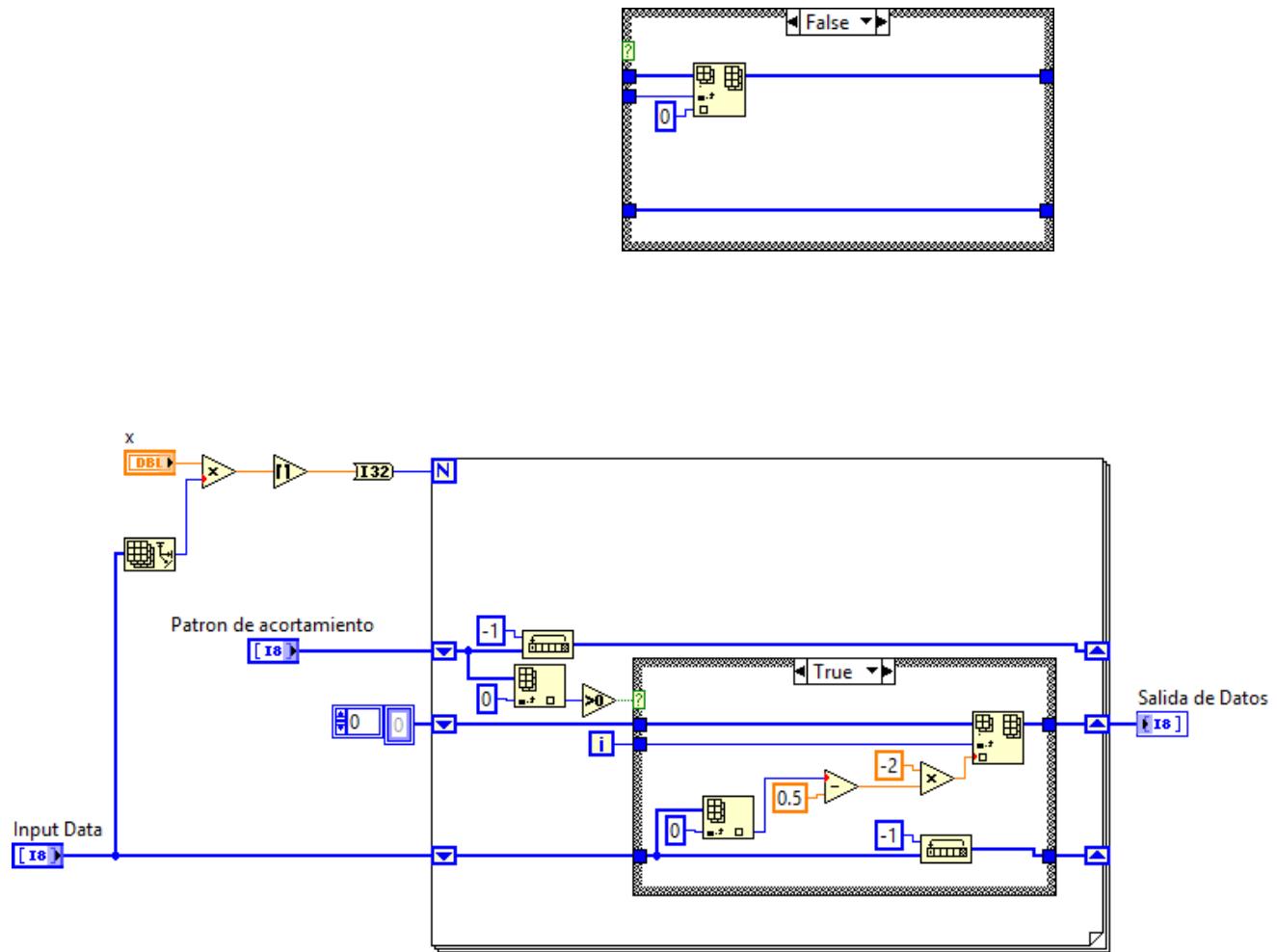
Caso 2: 16-QAM



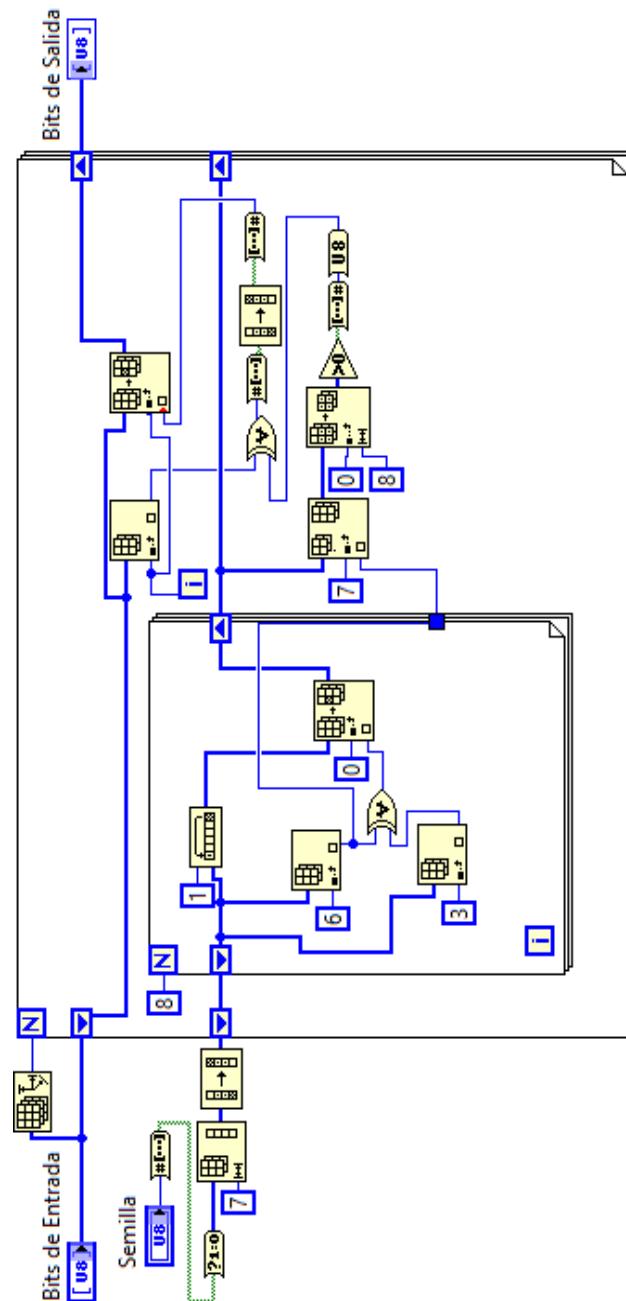
Caso 3: 64-QAM



ANEXO E. DEPUNCTURING



ANEXO F. DESCRAMBLER





ANEXO G. PROGRAMA PARA LA MEDICIÓN DEL BER (Matlab)

```
clear all
clc

#####%
%ANALISIS DE LAS MUESTRAS TRANSMITIDAS
a=fopen('ARCHIVO_tx','r');
b=fread(a);
c=6;h=1;
newline=find(b==10);
nfr=0;
for j=2:length(newline)-1
    if newline(j)-newline(j-1)==1
        if newline(j+1)-newline(j)==3
            if newline(j+2)-newline(j+1)==1
                if (b(newline(j)+1)==64)
                    if (b(newline(j)+2)==64)
                        nfr=nfr+1;
                    end
                end
            end
        end
    end
end
startline=zeros(nfr,1);

for j=2:length(newline)-1
    if newline(j)-newline(j-1)==1
        if newline(j+1)-newline(j)==3
            if newline(j+2)-newline(j+1)==1
                if (b(newline(j)+1)==64)
                    if (b(newline(j)+2)==64)
                        startline(h)=newline(j+2);
                        h=h+1;
                    end
                end
            end
        end
    end
end
d=length(startline);e=startline(2)-12;
b_vec=zeros(e,d);

for j=1:d
    for k=1:e
        c=c+1;
        b_vec(k,j)=b(c);
    end
    c=c+6;
end
end

b_tx=b_vec;
```



```
%#####ANALISIS DE LAS MUESTRAS RECIBIDAS#####
a=fopen('cancion_rx_bpsk12snr_prueba','r');
bp=fread(a);
d=zeros(length(bp),2);
h=1;k=1;d0=0;d1=0;
newline=find(bp==64);
nfr=0;
%Asignar valor del número de secuencia
for j=2:length(newline)-1
    if newline(j)-newline(j-1)==1
        if newline(j+1)-newline(j)==5
            if newline(j+2)-newline(j+1)==1
                if (bp(newline(j)+1)==10)
                    if (bp(newline(j)+4)==10)

bp(newline(j)+2)=((bp(newline(j)+2))*256)+bp(newline(j)+3);
d(k,1)=newline(j)+3;
d(k,2)=1;
k=k+1;
bp(newline(j)-1)='~';
bp(newline(j))='~';
bp(newline(j)+1)='=';
bp(newline(j)+4)='=';
bp(newline(j)+5)='~';
bp(newline(j)+6)='~';
end
end
end
end
end
d0=zeros(length(bp)-(length(find(d==0))/2),2);

for j=1:1:length(d0)
d0(j,1)=d(j,1);
d0(j,2)=d(j,2);
d1=d1+d0(j,2);
end
d0=sort(d0);
b=zeros(length(bp)-d1,1);
k=1;j=1;
for h=1:length(b)
    if k==d0(j,1)
        k=k+d0(j,2);
        if j<length(d0)
            j=j+1;
        end

    end
    b(h)=bp(k);
    k=k+1;
end

newline=find(b==10);
```



```
for j=2:length(newline)-1
    if newline(j)-newline(j-1)==1
        if newline(j+1)-newline(j)==3
            if newline(j+2)-newline(j+1)==1
                if (b(newline(j)+1)==64)
                    if (b(newline(j)+2)==64)
                        nfr=nfr+1;
                end
            end
        end
    end
end

startline=zeros(nfr,1);
h=1;

for j=2:length(newline)-1
    if newline(j)-newline(j-1)==1
        if newline(j+1)-newline(j)==3
            if newline(j+2)-newline(j+1)==1
                if (b(newline(j)+1)==64)
                    if (b(newline(j)+2)==64)
                        startline(h)=newline(j+2);
                        h=h+1;
                end
            end
        end
    end
end
end

dif=zeros(length(startline),1);
for j=2:length(startline)
    dif(j-1)=startline(j)-startline(j-1);
end

dif(length(startline))=length(b)-startline(length(startline))+6;
frame_length=mode(dif);
%
limite=150+frame_length;
%
garp=find(dif~=frame_length & dif < limite);
gar_pos=zeros(length(garp),1);
gar_length=zeros(length(garp),1);
garl=0;

for j=2:length(garp)
    garl=max(dif(garp(j)),dif(garp(j-1)));
end

for j=1:length(garp)
    gar_pos(j)=startline(garp(j))-5;
    gar_length(j)=dif(garp(j));
end

garbage=zeros(garl,length(garp));
i=0;
for j=1:1:length(gar_pos)
```



```
k=gar_pos(j)+6;
i=i+1;
for h=1:(gar_length(i)-6)
    garbage(h,j)=b(k);
    k=k+1;
end
end

newframe=find(dif==frame_length);
nfr2=length(newframe);
frame_length=frame_length-6;
b_vec=zeros(frame_length,nfr2);

for j=1:nfr2

b_vec(1:frame_length,j)=b(startline(newframe(j))+1:startline(newframe(j))+frame_length);
end

n_seq=b_vec(frame_length-3,:);
b_vecsize=size(b_vec);
%
b_frame=zeros(b_vecsize(1)-7,max(n_seq));
%
g=size(b_frame);

for j=1:nfr2
    b_frame(:,n_seq(j)+1)=b_vec(1:(g(1)),j);
end

#####
%COMPARACIÓN ENTRE MUESTRAS RECIBIDAS Y MUESTRAS TRANSMITIDAS
b_rx=b_frame;
b_rx2=zeros(size(b_tx));
WrBit=0;
n_Framenrx=0;
n_WrFrame=0;
n1=size(b_tx);
n2=size(b_rx);
b_tx2=zeros(max(n1(1),n2(1)),max(n1(2),n2(2)));
b_rx2=zeros(max(n1(1),n2(1)),max(n1(2),n2(2)));

for j=1:n1(2)
    for k=1:n1(1)
        b_tx2(k,j)=b_tx(k,j);
    end
end
for j=1:n2(2)
    for k=1:n2(1)
        b_rx2(k,j)=b_rx(k,j);
    end
end

b_tx=b_tx2;
b_rx=b_rx2;

nfr=zeros(1,2);
nfr(1)=max(n1(1),n2(1));
nfr(2)=max(n1(2),n2(2));

Wr_Frame=zeros(1,nfr(2));
```



```
NRx_Frame=zeros(1,nfr(2));
n3=size(garbage);
for j=1:nfr(2)
    if (b_rx(:,j)==0)
        WrBit=WrBit+nfr(1)*8;
        n_Framenrx=n_Framenrx+1;
        NRx_Frame(j)=1;
    else
        diff=find((b_tx(:,j)-b_rx(:,j))~=0);
        if length(diff)~=0
            n_WrFrame=n_WrFrame+1;
            Wr_Frame(j)=1;
        end
        for k=1:length(diff)
            m=dec2bin(b_tx(diff(k),j));
            n=dec2bin(b_rx(diff(k),j));

            while length(m)<8,
                m=strcat('0',m);
            end
            if length(m)>8
                A=regexp(m, '\w{1,8}', 'match');
                m=char(A(1));
            end

            while length(n)<8,
                n=strcat('0',n);
            end
            if length(n)>8
                A=regexp(n, '\w{1,8}', 'match');
                n=char(A(1));
            end

            WrBit=WrBit+length(find((m-n)~=0));
        end
    end
end
BER=WrBit/(nfr(1)*nfr(2)*8);
Wr_Frame=find(Wr_Frame==1);
NRx_Frame=find(NRx_Frame);
g=size(garbage);
h=1;
ctx=size(b_tx);
comp=zeros(length(gar_pos),length(NRx_Frame));
for j=1:length(gar_pos)
    i=0;
    for k=1:length(NRx_Frame)
        l=NRx_Frame(k);
        i=1+i;
        for h=1:n1(1)
            if garbage(h,j)==b_tx(h,l)
                comp(j,k)=comp(j,k)+1;
            end
        end
    end
end
threshold=0.3*nfr(1);
mej=zeros(length(gar_pos));
for j=1:length(gar_pos)
    if (length(find(comp(j,:)==max(comp(j,:))&max(comp(j,:))>threshold))==1)
        mej(j)=find(comp(j,:)==max(comp(j,:))&max(comp(j,:))>threshold);
    end
end
```



```
elseif (length(find(comp(j,:)==max(comp(j,:))&max(comp(j,:))>threshold))>1)
    t=find(comp(j,:)==max(comp(j,:))&max(comp(j,:))>threshold);
    mej(j)=t(1);
else
    mej(j)=0;
end

end

mej=mej(:,1);
for j=1:length(mej)
    if mej(j)~=0
        b_rx(:,NRx_Frame(mej(j)))=garbage(1:n1(1),j);
    end
end

b_rx2=zeros(size(b_tx));
WrBit=0;
n_Framenrx=0;
n_WrFrame=0;
Wr_Frame=0;
NRx_Frame=0;

for j=1:nfr(2)
    if (b_rx(:,j)==0)
        WrBit=WrBit+nfr(1)*8;
        n_Framenrx=n_Framenrx+1;
        NRx_Frame(j)=1;
    else
        diff=find((b_tx(:,j)-b_rx(:,j))~=0);
        if length(diff)~=0
            n_WrFrame=n_WrFrame+1;
            Wr_Frame(j)=1;
        end
        for k=1:length(diff)
            m=dec2bin(abs(b_tx(diff(k),j)));
            n=dec2bin(abs(b_rx(diff(k),j)));

            while length(m)<8,
                m=strcat('0',m);
            end
            if length(m)>8
                A=regexp(m, '\w{1,8}', 'match');
                m=char(A(1));
            end

            while length(n)<8,
                n=strcat('0',n);
            end
            if length(n)>8
                A=regexp(n, '\w{1,8}', 'match');
                n=char(A(1));
            end
            WrBit=WrBit+length(find((m-n)~=0));
        end
    end
end

BER=WrBit/(nfr(1)*nfr(2)*8);
Wr_Frame=find(Wr_Frame==1);
NRx_Frame=find(NRx_Frame);
```