



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

**Desarrollo e implementación de la modulación BPSK más silencios mediante
radios NI USRP.**

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA ELÉCTRICA

PRESENTA:
ING. RENÉ RAMÓN GALLEGO SALGADO

TUTOR PRINCIPAL
DR. JAVIER GÓMEZ CASTELLANOS
FACULTAD DE INGENIERIA

CIUDAD DE MÉXICO, OCTUBRE DE 2019



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO ASIGNADO

Presidente: Dr. José María Matías Maruri
Secretario: Dr. José Jaime Camacho Escoto
Vocal: Dr. Javier Gómez Castellanos
1er. Suplente: Dr. Daniel Enrique Ceballos Herrera
2do. Suplente: Dr. Víctor Rangel Licea

La tesis se realizó en la Facultad de Ingeniería, UNAM.

TUTOR DE TESIS:
Dr. Javier Gómez Castellanos

Agradecimientos

A mi familia...

... por todo su amor, paciencia y apoyo para poder alcanzar mis metas, agradecerles por mi formación como persona. A mis padres por sus sabios consejos y dedicar gran parte de sus vidas a educarme y ser el regalo más grande que me ha concedido la vida. A mi abuela Margarita y mi tío Romárico por todo su cariño y dejar una huella imperecedera en mi vida. A mi esposa Elizabet por estar ahí siempre que la necesité y demostrarme que no estaba solo, por arriesgarse conmigo en búsqueda de mi crecimiento profesional.

A mi tutor...

... por brindarme el apoyo y la paciencia para la realización de esta tesis, por sus enseñanzas, por permitirme ser parte de su grupo de trabajo.

A mis amigos...

... por el apoyo durante toda esta etapa.

A Octavio Alfonso Jaimes Botello...

... por su ayuda incondicional cuando la necesité, sin apenas conocerme.

A la Universidad Nacional Autónoma de México...

... por darme la oportunidad de crecer profesionalmente, por esa gran formación educativa que me ha proporcionado. Gracias por dejarme sentir el orgullo de ser UNAM.

A mis profesores...

... por ser pilares fundamentales de mi formación, los que me forjaron como ingeniero en la Universidad de Oriente en Cuba, los del posgrado de ingeniería de la UNAM por todo el aporte a mi formación, me siento orgulloso de formar parte de sus alumnos. Al doctor Matías por confiar en mí.

Al Consejo Nacional de Ciencia y Tecnología...

... por el apoyo económico otorgado para poder realizar la Maestría.

A la DGAPA-UNAM...

... por el apoyo otorgado al proyecto de investigación PAPIIT IN117017.

Declaración de autenticidad

Por la presente declaro que, salvo cuando se haga referencia específica al trabajo de otras personas, el contenido de esta tesis es original y no se ha presentado total o parcialmente para su consideración para cualquier otro título o grado en esta o cualquier otra universidad. Esta tesis es resultado de mi propio trabajo y no incluye nada que sea el resultado de algún trabajo realizado en colaboración, salvo que se indique específicamente en el texto.

Ing. René Ramón Gallego Salgado. Ciudad de México, 13 de octubre de 2019.

Actualmente el uso de las redes de datos y el acceso a Internet se ha masificado, así como las diversas tecnologías empleadas para la interconexión a estas redes. Uno de los medios físicos de comunicación más empleado es el acceso inalámbrico, el cual permite al usuario ubicarse en cualquier lugar, disfrutar de movilidad y además disponer de tasas de transferencia de información aceptables. Sin embargo, el ancho de banda limitado y que los dispositivos en ocasiones no poseen ninguna fuente de alimentación estándar, sino que dependen de baterías, hace necesario concebir métodos que nos permitan obtener lo mejor de estos recursos. Las redes de sensores inalámbricos son un caso especial de tecnología inalámbrica, donde la optimización de energía juega un papel determinante en el desempeño de la red. El elemento que contribuye en mayor medida al consumo energético de un nodo sensor es el módulo de comunicaciones. Debido a lo anterior, la presente tesis se enfoca en el desarrollo e implementación de un nuevo esquema de codificación-modulación, la modulación por desplazamiento de fase binaria más silencios (BPSK^{+S}) y su empleo en redes inalámbricas, especialmente redes de sensores y dispositivos del Internet de las cosas. La nueva modulación BPSK^{+S} , aunada a la información binaria propia de BPSK agrega un tercer estado llamado silencio (sin transmisión) que tiene significado en el nodo receptor, además de establecer un método de codificación que reduce el número de bits a modular. Se emplean los radios NI USRP para simular un sistema inalámbrico que utiliza este esquema de modulación, y de esta manera realizar pruebas enfocadas a la caracterización del desempeño práctico del sistema: BER contra SNR, sincronización, tamaño óptimo de la ventana y bloque de codificación, contención de la propagación de errores en la decodificación y estimación del ahorro energético.

Currently the use of data networks and internet access has become widespread, as well as the various technologies used to interconnect these networks. One of the most used physical medium of communication is wireless access, which allows the user to be located anywhere, enjoy mobility and also have acceptable information transfer rates. However, bandwidth is limited and devices sometimes do not have any standard power supply, but depend on batteries, so it is necessary to conceive methods that allow us to get the best of these resources. Wireless sensor networks are a special case of wireless technology, where the optimization of energy plays a decisive role in the performance of the network. The element that contributes the most to the energy consumption of a sensor node is the communications module. Due to the above, this thesis focuses on the development and implementation of a new coding-modulation scheme, the binary phase shift modulation plus silences (BPSK^{+S}) and its use in wireless networks, especially sensor networks and devices of the Internet of things. The new modulation BPSK^{+S}, together with BPSK's own binary information, adds a third state called silence (without transmission) that has meaning in the receiver node, in addition to establishing a coding method that reduces the number of bits to be modulated. The NI USRP radios are used to simulate a wireless system that uses this modulation scheme, and in this way perform tests focused on the characterization of the practical performance of the system: BER vs SNR, synchronization, optimal window size and coding block, containment of the propagation of errors in decoding and estimation of energy saving.

Índice de figuras	XI
Índice de tablas	XIII
Lista de acrónimos	XIV
1. Introducción	1
1.1. Antecedentes	1
1.2. Definición del problema	2
1.3. Objetivos	2
1.4. Contribuciones y relevancia	3
1.5. Metodología.	3
1.6. Estructura de la tesis	3
2. Redes de sensores inalámbricos y el consumo energético	5
2.1. Redes de sensores inalámbricos	5
2.2. Aplicaciones	6
2.2.1. Medio ambiente	6
2.2.2. Aplicaciones militares y entornos de alta seguridad	7
2.2.3. Eficiencia energética	7
2.2.4. Domótica y salud	7
2.2.5. Industriales y vehículos	8
2.3. Arquitectura.	8
2.3.1. Capa física	8
2.3.1.1. Elección del ancho de banda	9
2.3.1.2. Esquema de modulación.	10
2.4. Estándares	10
2.4.1. Estándar IEEE 802.15.4.	10
2.4.2. ZigBee	12
2.4.3. Otros estándares	13
2.5. Hardware.	14
2.5.1. Motes con protocolo ZigBee	15
2.5.1.1. Xbee	16
2.5.1.2. ETRX357	16
2.5.1.3. JN5139-001-MXX	17
2.6. Optimización del consumo energético en redes de sensores inalámbricos.	17

3.	Modulación digital y la técnica de codificación-modulación BPSK^{+S}	21
3.1.	Modulación	21
3.1.1.	Modulación digital	22
3.1.1.1.	Modulación digital pasa banda.	22
3.2.	Parámetros de las modulaciones digitales.	23
3.2.1.	Eficiencia espectral.	23
3.2.2.	Ruido y relación señal a ruido	24
3.2.3.	Tasa de error de bit y probabilidad de error.	25
3.2.4.	Tasa de error de modulación	27
3.2.5.	Diagrama de constelación.	28
3.3.	Modulaciones de referencia para el desarrollo de BPSK ^{+S}	29
3.3.1.	Modulación OOK.	29
3.3.2.	Modulación BPSK	31
3.4.	Técnica de codificación-modulación BPSK ^{+S} (propuesta).	33
3.4.1.	Codificación BPSK ^{+S}	33
3.4.1.1.	Error de bit en avalancha	35
3.4.2.	Modulación BPSK ^{+S}	36
3.4.3.	Probabilidad de error de bit en BPSK ^{+S}	37
4.	Implementación de BPSK^{+S} en radios NI USRP	41
4.1.	Herramientas de <i>hardware</i>	41
4.1.1.	NI USRP 2932	42
4.1.1.1.	Arquitectura	42
4.1.1.2.	Antenas.	44
4.2.	Herramientas de <i>software</i>	45
4.2.1.	LabVIEW.	45
4.2.1.1.	LabVIEW y los radios NI USRP	47
4.2.2.	MATLAB.	49
4.2.2.1.	Entorno de programación.	50
4.3.	Implementación de BPSK ^{+S}	51
4.3.1.	Transmisor BPSK ^{+S}	51
4.3.1.1.	Bloque configuración NI USRP 2932	53
4.3.1.2.	Bloque conversor de texto a bits.	53
4.3.1.3.	Bloque codificador BPSK ^{+S}	54
4.3.1.4.	Bloque modulador BPSK ^{+S}	56
4.3.2.	Receptor BPSK ^{+S}	56
5.	Experimentos y resultados	61
5.1.	Metodología de los experimentos	61
5.2.	Selección del canal y SNR	62
5.3.	Validación de la implementación de BPSK ^{+S}	64
5.4.	BER en BPSK ^{+S}	65
5.5.	Variación ventana de codificación	68
5.6.	Variación del tamaño de bloque de codificación	71
5.6.1.	Tamaño de bloque y efecto BERA.	73
5.7.	Número de estados S o silencios seguidos permitidos	74
5.8.	Campo de acción de BPSK ^{+S}	75
6.	Conclusiones	79

A. Código/Manuales/Publicaciones	81
A.1. Código función de probabilidad de error de bit (MATLAB)	81
A.2. Código función codificador BPSK ^{+S} (MATLAB)	82
A.3. Código función decodificador BPSK ^{+S} (MATLAB)	85
A.4. La función error.	88
A.5. Especificaciones de módulos RF Xbee/Xbee-PRO ZB	89
A.6. Diagrama de maqueta	91
Bibliografía	92

Índice de figuras

2.1.	Estructura y componentes de un nodo sensor.	5
2.2.	Infraestructura de una red inalámbrica de sensores	8
2.3.	Estructura de la trama de capa física IEEE 802.15.4	12
2.4.	Protocolos de IEEE 802.15.4	12
2.5.	Diagrama de la estructura de una red ZigBee.	13
2.6.	Módulo ZigBee Xbee.	16
2.7.	Módulo ZigBee ETRX357.	16
2.8.	Módulos ZigBee JN5139-001-MXX	17
2.9.	Domínios del consumo energético en WSN.	17
3.1.	Función de densidad de probabilidad gaussiana.	26
3.2.	Región de decisión y probabilidad de error de símbolo.	27
3.3.	Representación gráfica del MER	28
3.4.	Representación de un símbolo en el diagrama de constelación	28
3.5.	Forma de onda de salida de un modulador OOK para $M=2$	29
3.6.	Diagrama de constelación para un modulador OOK	30
3.7.	Región de decisión en modulación OOK.	30
3.8.	Relación de fase de salida en función del tiempo para un modulador BPSK.	31
3.9.	Diagrama de constelación para un modulador BPSK.	32
3.10.	Región de decisión en modulación BPSK, para $n_c = 2$	32
3.11.	Bloque de m bits a codificar.	33
3.12.	Registros del codificador BPSK ^{+S}	34
3.13.	Registros del decodificador BPSK ^{+S}	35
3.14.	Efecto BERA	36
3.15.	Forma de onda de salida de un modulador BPSK ^{+S}	37
3.16.	Diagrama de constelación para un modulador BPSK ^{+S}	37
3.17.	Función de densidad de probabilidad para los tres estados de BPSK ^{+S}	38
3.18.	Región de decisión en modulación BPSK ^{+S}	38
3.19.	Probabilidad de error de bit en modulaciones OOK, BPSK y BPSK ^{+S}	39
4.1.	Radio NI USRP 2932	42
4.2.	Diagrama en bloques de NI USRP 2932	43
4.3.	Tarjeta hija SBX.	44
4.4.	Antenas usadas por el NI USRP 2932	44
4.5.	Panel frontal y diagrama de bloques.	46

4.6.	MATLAB <i>script node</i> y paleta de funciones	47
4.7.	Paleta NI-USRP en LabVIEW.	48
4.8.	Bloques de inicio de sesión en el transmisor y el receptor.	48
4.9.	Bloque de configuración de señal.	48
4.10.	Bloque inicio de recepción de señal de RF.	49
4.11.	Bloques de intercambio de datos	49
4.12.	Bloques de abordar y cerrar sesión.	49
4.13.	Interfaz de MATLAB.	50
4.14.	Panel Frontal transmisor BPSK ^{+S} en LabVIEW	51
4.15.	Diagrama en bloques transmisor BPSK ^{+S} en LabVIEW	52
4.16.	Parámetros de configuración NI USRP 2932.	53
4.17.	Convertor de texto a bits	54
4.18.	Codificador BPSK ^{+S} en LabVIEW	55
4.19.	Modulador BPSK ^{+S} en LabVIEW	56
4.20.	Diagrama en bloques receptor BPSK ^{+S} en LabVIEW.	57
4.21.	Panel Frontal receptor BPSK ^{+S} en LabVIEW	58
4.22.	Parámetros de configuración receptor NI USRP 2932.	58
5.1.	Espectro de potencia del canal libre.	62
5.2.	Amplitud de la potencia espectral.	63
5.3.	Constelación de BPSK ^{+S} en transmisión y recepción	64
5.4.	Mensaje codificado y decodificado en BPSK ^{+S} : a) en transmisión y b) en recepción.	65
5.5.	BER contra SNR en BPSK ^{+S} teórico y real.	66
5.6.	Bloque de AWGN.	66
5.7.	BER contra SNR en BPSK ^{+S} teórico y simulado.	67
5.8.	Relación tamaño de ventana de codificación y número de estados transmitidos totales	68
5.9.	Relación tamaño de ventana de codificación y número de estados transmitidos (0 y 1) respecto al total de bits de la secuencia a codificar (856 bits)	69
5.10.	Relación tamaño de ventana de codificación y porcentaje de estados transmitidos (0 y 1) respecto al total de estados transmitidos.	69
5.11.	Relación tamaño de ventana de codificación y tasa de errores para SNR=16 dB	70
5.12.	Porcentaje de estados transmitidos para ventana $w=1$: a) totales b) estados S.	70
5.13.	Relación tamaño de bloque de codificación y número de estados transmitidos totales.	71
5.14.	Relación tamaño de bloque de codificación y número de estados transmitidos (0 y 1).	72
5.15.	a) Número mínimo de tamaño de bloque y b) porcentaje de estados (0 y 1) transmitidos por cada secuencia.	72
5.16.	Efectos de los errores en el canal con $m=856$	73
5.17.	Efectos de los errores en el canal con $m=24$	74
5.18.	Porcentaje de estados de relleno utilizados	75
5.19.	Módulo Xbee Pro S2C y antena dipolo A24-HASM-450	76
5.20.	Radio de cobertura modulaciones BPSK ^{+S} y BPSK	78

Índice de tablas

2.1.	Resumen capa física estándar 802.15.4	11
2.2.	Resumen de datos de algunas plataformas disponibles en el mercado para su empleo en WSN.	15
3.1.	Resumen de parámetros de las modulaciones digitales	23
3.2.	Comparación de presencia de errores en diversos esquemas de modulación digital.	25
4.1.	Principales funciones de LabVIEW para comunicación con NI-USRP 2932	48
4.2.	Estructura del paquete final a transmitir.	55
5.1.	Objetivos de los experimentos a realizar	61
5.2.	Niveles de potencia de transmisión y SNR.	63
5.3.	Valores de BER transmisión real	65
5.4.	Valores de BER transmisión simulada.	67
5.5.	Valores de BER contra tamaño de bloque para SNR=16 dB.	74
5.6.	Modelo de propagación	76

Lista de acrónimos

ACK	Acknowledgement
ADC	Analogic-Digital Converter
AODV	Ad-hoc on-Demand Distance Vector
ASCII	American Standard Code for Information Interchange
ASK	Amplitude Shift Keying
AWGN	Additive White Gaussian Noise
BERA	Bit Error Rate Avalanche
B-MAC	Berkeley Media Access Control
BPSK	Binary Phase Shift Keying
BPSK^{+S}	Binary Phase Shift Keying Plus Silence
CCA	Clear Channel Assessment
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance
CSS	Chirp Spread Spectrum
CTS	Clear to Send
DAC	Digital-Analogic Converter
DDC	Digital Down Converter
DH-AODV	Directional Hierarchical Ad-hoc on-Demand Distance Vector
DSMAC	Dynamic Sleeping Media Access Control
DSSS	Direct Sequence Spread Spectrum
EEPROM	Electrically Erasable Programmable Read-Only Memory
EIRP	Effective Isotropic Radiated Power
FEC	Forward Error Correction
FFD	Full Function Device
FFT	Fast Fourier Transform
FPGA	Field Programable Gate Array
FSK	Frequency Shift Keying
GFSK	Gaussian Frequency Shift Keying
GPS	Global Positioning System
GSM	Global System for Mobile Communications
HART	Highway Addressable Remote Transducer
IBM	International Business Machines Corporation
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol

ISI	Intersymbol Interference
ISM	Industrial, Scientific and Medical
IoT	Internet of Things
LabVIEW	Laboratory Virtual Instrument Engineering Workbench
LED	Light Emitting Diode
LNA	Low Noise Amplifier
LoRaWAN	Low Power Wide Area Network
NASA	National Aeronautics and Space Administration
NXP	Next eXPerience
NI USRP	National Instruments Universal Software Radio Peripheral
MAC	Media Access Control
MATLAB	Matrix Laboratory
MEMS	Microelectromechanical Systems
MER	Modulation Error Ratio
OFDM	Orthogonal Frequency Division Multiplexing
OOK	On-Off Keying
O-QPSK	Offset Quadrature Phase Shift Keying
OSI	Open System Interconnection
PDF	Probability Density Function
PMU	Power Management Unit
PSDU	Physical Service Data Unit
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
RAM	Random Access Memory
RFD	Reduced Function Device
RFID	Radio Frequency Identification
RTS	Request to Send
SDR	Software Defined Radio
SMA	SubMiniature version A
SMAC	Sensor Medium Access Control
SNR	Signal to Noise Ratio
TDMA	Time Division Multiplexed Access
T-MAC	Timeout Medium Access Control
UIT-T	International Telecommunication Union
UWB	Ultra-wide Band
VI	Virtual Instrument
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network
WirelessHART	Wireless Highway Addressable Remote Transducer
WiseMAC	Wireless Sensor Media Access Control

1.1. Antecedentes

Hoy día, las comunicaciones inalámbricas son una necesidad básica en la sociedad. Hay una tendencia creciente en utilizar enlaces de datos inalámbricos, tanto en aplicaciones que requieren movilidad como en las fijas. La distancia que han de abarcar estos enlaces es muy distinta dependiendo de las aplicaciones y puede ir desde distancias muy cortas, de pocos metros, hasta bastantes kilómetros.

Para comunicarse de forma inalámbrica existen dos alternativas principales. La primera es la comunicación vía óptica en el espectro visible o infrarrojo, mediante LEDs (*light-emitting diode*) o láseres. La segunda alternativa es la comunicación por radio. Esta última ofrece un amplio rango de velocidades de datos y una gran variedad de modulaciones, desde las más tradicionales: ASK (*amplitude shift keying*) y FSK (*frequency shift keying*) hasta las más complejas: OFDM (*orthogonal frequency division multiplexing*), entre otras. Las modulaciones digitales son ampliamente utilizadas en los sistemas de comunicación por radio, ya que permiten grandes velocidades de transmisión, presentan mayor inmunidad al ruido con respecto a sistemas analógicos, y manejan con mayor facilidad las tasas de error y a su vez tratan de corregirlas.

Existen varias formas de modular digitalmente una portadora, desde la más simple que es sencillamente prender y apagar la portadora con cada bit a transmitir, conocida como OOK (*on-off keying*); esquemas donde en lugar de la amplitud se varía la frecuencia con cada bit a transmitir: FSK, otros donde se cambia la fase de la señal: PSK (*phase shift keying*) y esquemas más complejos, donde se manipula simultáneamente la amplitud y la fase de la señal portadora: QAM (*quadrature amplitude modulation*).

Por su simplicidad y alta inmunidad al ruido, la modulación PSK y en especial BPSK (*binary phase shift keying*) es utilizada entre otras muchas aplicaciones en redes de sensores, unos sistemas que tienen un interés máximo cuando se realizan inalámbricamente: WSN (*wireless sensor network*). [1]

Un aspecto fundamental en el diseño y despliegue de WSNs es el consumo de energía de los nodos. Las soluciones tradicionales basadas en el empleo de baterías, o bien conectar los nodos a la red eléctrica (cuando esté disponible) son soluciones viables sólo para un conjunto limitado de usos. Para una aceptación amplia de estas redes, una vez se ha emplazado un nodo en ellas, estas deberían poder operar de manera no atendida, mientras que el recambio de la batería puede ser una opción inviable o poco práctica. Las limitaciones de energía de las WSNs dan lugar a una serie de principios de diseño distintos a los que habitualmente existen en las redes tradicionales. El elemento que contribuye en mayor medida al consumo de energía de un nodo sensor es el módulo de comunicaciones (tanto la transmisión como la recepción de señales). El procesado de datos también consume energía, pero en menor medida. De ello se deduce que es preferible procesar la información de que se dispone si con ello se ahorran bits a transmitir [2].

En esta investigación se desarrolla la introducción en la modulación BPSK de un tercer estado (un silencio), que también representa bits de datos: BPSK^{+S} (*binary phase shift keying plus silence*). Al transmitir bits o información con los silencios, se busca tener un ahorro en la energía utilizada por el transceptor, pues para transmitir un silencio no se irradia ninguna señal. Además del empleo de una estrategia de codificación que

permite la reducción del número de bits a modular. Lo que representa una propuesta atractiva para este tipo de dispositivos que tienen una severa limitación de energía disponible.

En el desarrollo de una nueva modulación, son de vital importancia la definición de las especificaciones de operación del hardware a utilizar, la recepción y transmisión de datos, la definición de la topología y la implementación de ciertos procedimientos de señalización y codificación. Actualmente *National Instruments* ha desarrollado una plataforma que intenta crear un enlace sencillo entre la teoría y la implementación práctica de los sistemas de comunicación por radio: los transceptores NI USRP (*national instruments universal software radio peripheral*). Estos equipos son dispositivos flexibles y potentes, que mediante el uso de softwares como MATLAB (*MATrix LABORatory*) y LabVIEW (*laboratory virtual instrument engineering workbench*) son capaces de generar prototipos de comunicación por radiofrecuencia de una forma sencilla [3]; siendo una herramienta útil para la simulación, el estudio y desarrollo de los sistemas inalámbricos.

1.2. Definición del problema

Las WSN han sido reconocidas como la tecnología más emergente e interesante para el desarrollo de IoT (*Internet of Things*). Lo que ha permitido aumentar su popularidad en la investigación industrial y académica [4].

Los nodos sensores dentro de una WSN deben cumplir tres requisitos imprescindibles: pequeñas dimensiones físicas para poder ser colocados en cualquier lugar con facilidad; necesitan poder comunicarse para enviar y recibir información y todo combinado con el hecho de ser muy eficientes energéticamente para lograr ser autónomos [5]. Precisamente esta necesidad de eficiencia energética dada sus pequeñas dimensiones, los hacen estar equipados con una fuente de alimentación limitada. Esto representa el principal problema de las WSN, lo cual provoca que su diseño se enfoque en la búsqueda de estrategias para gestionar adecuadamente este vital recurso.

Dentro de las soluciones exploradas encontramos la piezoeléctrica, donde se extrae energía de movimientos o vibraciones. Se ha aplicado en suelos para extraer energía de las pisadas, o por ejemplo en [6], donde se utilizan celdas de material piezoeléctrico para aprovechar las corrientes de agua en mares y ríos en la generación de unos pocos milivatios, pero suficientes para cargar la batería de un nodo sensor.

Desde el punto de vista del *firmware*, una técnica que a veces se utiliza es tener un manejo variable de la frecuencia de reloj, así como la disminución del voltaje de alimentación del microprocesador durante períodos de baja actividad; aunque a veces compromete el rendimiento del sistema.

Es conocido que los tres primeros niveles de la pila de protocolos en WSN tienen un impacto fundamental en el consumo energético [7]. En cuanto al nivel físico y de acceso al medio encontramos una de las primeras propuestas: SMAC (*sensor medium access control*) [8]. Este protocolo es sencillo: utiliza un esquema despierto/dormido donde el período despierto se subdivide en tres ranuras temporales (*slots*). Muchas otras variantes de protocolo de acceso al medio han sido propuestas: BMAC (*berkeley media access control*) [9], WiseMAC (*wireless sensor media access control*) [10] y DSMAC (*dynamic sleeping media access control*) [11].

En conclusión, el estudio de la eficiencia energética en las redes de sensores inalámbricos es un problema de vital importancia en su área de investigación y cuya mejora ampliará los límites de las aplicaciones que se pueden cubrir con esta tecnología. Poco o nada se ha realizado en cuanto al desarrollo de mejoras desde el punto de vista de la modulación.

1.3. Objetivos

El objetivo principal de esta tesis es desarrollar e implementar la nueva técnica de codificación-modulación: $BPSK^{+S}$, que buscará optimizar el consumo energético en los nodos sensores pertenecientes a una WSN. Se caracterizarán los aspectos teóricos fundamentales de la nueva modulación: diagrama de constelación, BER y SNR. Se desarrollará una implementación utilizando los productos desarrollados por *National Instruments*: NI-USRP programados con las plataformas: LabVIEW y MATLAB. Se llevará a cabo una evaluación del desempeño práctico del sistema: BER contra SNR, sincronización, tamaño óptimo de la ventana y bloque de codificación, contención de la propagación de errores en la decodificación, así como la estimación del ahorro energético, estableciendo conclusiones.

1.4. Contribuciones y relevancia.

Hoy día, las redes inalámbricas de sensores son un innovador y atractivo campo de investigación debido a las disímiles aplicaciones que presentan. En [12], [13], se muestra a las redes de sensores inalámbricas como una tecnología fundamental del IoT, junto a otras tecnologías como la RFID (*radio frequency identification*), sistemas de *middleware* y la computación en la nube. No obstante, los nodos sensores poseen limitaciones en cuanto a: alcance de transmisión; capacidad de procesamiento; capacidad de almacenamiento, y suministro de energía. Precisamente el suministro de energía es un aspecto determinante al momento de estimar el tiempo de vida de la red y por ello es un factor de diseño en aplicaciones WSN. Dadas estas limitaciones, las investigaciones sobre WSN centran sus esfuerzos en estudiar y generar soluciones para que los nodos sensores hagan uso óptimo de sus recursos energéticos, [14], [15]. De este tipo de investigaciones nacieron protocolos y tecnologías de comunicación de bajo consumo, como LoRaWAN (*low power wide area network*) [16] y ZigBee [17], este último basado en el estándar IEEE 802.15.4 [18].

En este trabajo de tesis se busca desarrollar un método de codificación-modulación que disminuirá el número de bits a modular y por tanto que serán transmitidos, así como la introducción en la modulación BPSK de un tercer estado llamado silencio (sin transmisión) que tendrá significado en el nodo receptor. Con ello, se evaluará bajo qué circunstancias y hasta qué punto es viable su utilización en la optimización del consumo energético de los nodos sensores aplicados a WSN y dispositivos del Internet de las cosas.

1.5. Metodología

El presente trabajo se compone de una recopilación y desarrollo teórico seguido de un estudio práctico. Primeramente, se realizará una investigación detallada de las propuestas actuales relacionadas con la optimización energética de los nodos sensores pertenecientes a una red de sensores inalámbricos. Se desarrollará una propuesta de solución a tan importante problemática, basada en la introducción de un nuevo estado a la modulación BPSK unido a una estrategia de codificación que reducirá el número de bits a modular.

En la parte experimental se utilizarán los productos desarrollados por *National Instruments*: radios NI-USRP; que facilitarán la implementación y evaluación de la nueva técnica de codificación-modulación, permitiendo establecer conclusiones sobre los valores más adecuados de tamaño de bloque y ventana de codificación, número de silencios seguidos permisibles antes de afectar la sincronización del sistema y estimación del ahorro energético.

1.6. Estructura de la tesis

Se divide el trabajo en 6 capítulos. En el capítulo 2 estudiará el estado del arte de las redes de sensores en cuanto a consumo de energía en ellas y en los nodos sensores en particular, incluyendo los trabajos relacionados. En el capítulo 3 se ofrece una descripción de las bases teóricas de las modulaciones digitales y de la técnica de codificación-modulación BPSK^{+S}, así como su funcionamiento. En el capítulo 4 se desarrolla una introducción del equipo utilizado, con una descripción de la arquitectura, componentes y características de los radios NI USRP 2932 y los softwares LabVIEW y MATLAB, además de la explicación detallada de la implementación realizada. El capítulo 5 contiene los resultados de los experimentos: análisis del BER y demás parámetros de las modulaciones y la codificación. Finalmente, el capítulo 6 muestra las conclusiones generales de la tesis en cuanto a la implementación y desarrollo de la modulación BPSK^{+S} en los radios NI USRP y su contribución al ahorro energético de los nodos sensores de una WSN.

Redes de sensores inalámbricos y el consumo energético

Las tecnologías de redes inalámbricas han tenido un rápido desarrollo en los últimos años. El avance más interesante ha sido en las redes de sensores inalámbricos debido a sus múltiples aplicaciones, en distintos sectores. Los principales analistas tecnológicos, valoran a las WSN como una de las opciones de futuro más prometedoras. Fabricantes como Microsoft, Intel, IBM, Motorola y *Texas Instruments*, por solo citar algunos, han lanzado líneas de investigación en esta tecnología.

En este capítulo se tratan los conceptos teóricos principales y las aplicaciones de las redes de sensores inalámbricos mediante varios ejemplos representativos, haciendo especial énfasis en la gestión y optimización del consumo de energía en ellas y en los nodos sensores en particular.

2.1. Redes de sensores inalámbricos

Debido a los grandes avances en los sistemas microelectromecánicos conocidos como: MEMS (*microelectromechanical systems*), las comunicaciones inalámbricas y la electrónica digital, se han desarrollado nodos sensores multifuncionales, de bajo coste y consumo energético, con un reducido tamaño y que se comunican inalámbricamente en diferentes distancias. Estos nodos conocidos como: motes, están equipados con uno o más sensores, un procesador, una memoria, una fuente de alimentación, un transceptor radio y un actuador¹, permiten el diseño de WSNs basadas en la colaboración de un gran número de nodos para desempeñar una tarea determinada [5]. Ver Figura 2.1.

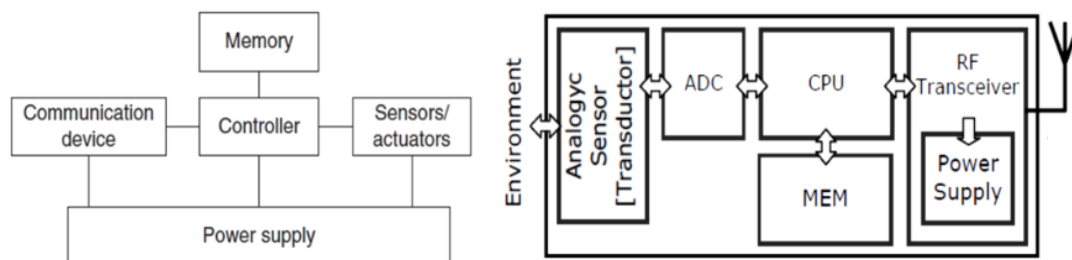


Figura 2.1: Estructura y componentes de un nodo sensor [19].

El sector de estandarización en telecomunicaciones de la ITU-T (*international telecommunication union*), define una red de sensores como una red compuesta por nodos sensores interconectados, que intercambian datos detectados, usando comunicación inalámbrica o cableada [20], [21].

¹ Un actuador es un dispositivo electromecánico que se puede usar para controlar diferentes componentes en un sistema. En un nodo sensor, los actuadores pueden actuar sobre los sensores, mover los nodos o monitorizar la energía en un nodo.

Las WSNs suponen una considerable mejora sobre los sensores tradicionales, que eran desplegados de una de las siguientes maneras [5]:

1. Los sensores eran colocados lejos del fenómeno real bajo estudio. En esta aproximación, se requerían sensores grandes que usasen algunas técnicas complejas para distinguir los objetivos del ruido presente en el ambiente.
2. Los sensores eran colocados sobre el fenómeno real bajo estudio, pero las posiciones de éstos y la topología de red se fijaban en una fase previa al despliegue. Además, los nodos transmitían, de forma periódica, las medidas tomadas del fenómeno a un nodo central donde se realizaba el procesamiento de datos.

En cambio, una WSN tiene poca o ninguna infraestructura y, generalmente, está constituida por un conjunto de cientos a miles de dispositivos autónomos, llamados nodos sensores, que están distribuidos físicamente en un área geográfica para monitorizar condiciones físicas o ambientales, con capacidad de almacenar y comunicar datos en una red de forma inalámbrica [22].

Los nodos pueden organizarse de forma automática en racimos o grupos, o bien colaborar para completar una tarea ordenada por el usuario. Además, la posición de estos nodos no necesita ser predefinida.

Existen dos tipos de WSNs: estructuradas y no estructuradas. La no estructurada consiste en una red de sensores que contiene una colección densa de nodos, pudiendo ser desplegados de una manera aleatoria en el entorno. Una vez desplegada, la red queda desatendida realizando las funciones para las que se diseñó. El mantenimiento de funciones de red como la conectividad y fallos en la detección es difícil debido a la gran cantidad de nodos presentes. Por otro lado, en las WSNs estructuradas, todos o algunos de los nodos son colocados en posiciones predefinidas. La ventaja de este tipo de redes es que los nodos pueden ser desplegados con un menor coste de gestión y mantenimiento de red; además se necesitan menos nodos en el despliegue ya que éstos se colocan en posiciones específicas para proporcionar la cobertura deseada, mientras que en despliegues aleatorios pueden quedar regiones sin cobertura.

Dada la gran variabilidad de usos y el ambiente donde son desplegadas estas redes: sobre la superficie de la tierra, debajo de la superficie terrestre o bajo el agua, una WSN afronta desafíos y restricciones diferentes. Según [5], existen cinco tipos fundamentales: terrestres, subterráneas, submarinas, multimedia y móviles.

2.2. Aplicaciones

El interés por las redes de sensores inalámbricos no ha parado de crecer en la medida que la tecnología ha hecho posible su realización. Su éxito radica en las innumerables posibilidades de empleo en forma de aplicaciones, las cuales pueden dividirse en dos grandes categorías [5]: monitorización y seguimiento.

Las aplicaciones de monitorización incluyen: la monitorización ambiental en entornos de interior y exterior, de la salud y bienestar de las personas, de energía, de ubicación, de inventario, la automatización de procesos y sistemas de producción, y la monitorización sísmica y estructural. Por otro lado, las aplicaciones de seguimiento engloban el seguimiento de objetos, animales, personas y vehículos.

En los siguientes apartados se mostrarán algunos ejemplos representativos de algunas aplicaciones prácticas en las que se han utilizado las redes de sensores.

2.2.1. Medio ambiente

El ámbito del medio ambiente quizás haya sido uno de los primeros beneficiados por las redes de sensores. Por ejemplo, se han utilizado redes de sensores para monitorizar la actividad sísmica de volcanes activos [23]. En esta aplicación se utilizaron 16 nodos *Tmote Sky* a lo largo de la cordillera de un volcán; dichos nodos iban retransmitiendo los datos mediante un protocolo multisaltos hasta una estación base que retransmitía los datos a gran distancia.

En aplicaciones agrícolas se puede comentar la monitorización de viñedos que hacen en [24]. En esta aplicación se dispone de un nodo central y varios nodos terminales que recogen datos ambientales y agrícolas mediante sensores de humedad de hoja y humedad del suelo.

También se han desarrollado varios proyectos para la detección temprana de incendios en bosques [25] [26]. Estas aplicaciones detectan incendios antes que otros métodos basados en satélites. En [25] los nodos sensan parámetros como la temperatura y humedad, los cuales retransmiten y son utilizados en los nodos centrales mediante algoritmos basados en redes neuronales artificiales para detectar la presencia de incendios. En [26] añaden detectores de humo, utilizan una suma ponderada de los parámetros medidos para discernir si hay fuego o no y presentan los datos y alarmas vía web.

En [27] desarrollan un sistema eficiente de control de regadíos basado en una matriz de sensores. La información de estos sensores corresponde a la humedad en diferentes profundidades del suelo, la cual es transmitida a través de múltiples saltos en la red a un nodo central con conexión 3G. La información recopilada ayudará a un operario a tomar la decisión del momento preciso para el regado de las plantaciones.

2.2.2. Aplicaciones militares y entornos de alta seguridad

También ha habido propuestas para utilizar las redes de sensores en el campo de batalla. Por ejemplo, en [28] utilizan una red agrupada en *clústers* para sensar señales acústicas y magnéticas que puedan dar lugar a detectar y localizar vehículos y movimiento de tropas en tiempo real.

En labores defensivas, en [29] proponen una red de sensores robusta y de baja latencia para la detección de posibles ataques contra instalaciones estratégicas o civiles. Para ello se plantean el escenario de una infraestructura de un metro de superficie y monitorizan vibraciones, temperaturas, ruido, olor, etc. para detectar infracciones en la seguridad.

Por otro lado, otro ámbito de aplicación es en lugares con un alto nivel de seguridad, tales como centrales nucleares. En [30] se hace un análisis de los requisitos y la aplicación de las redes de sensores inalámbricas aplicadas a la monitorización de centrales de generación de energía nuclear. Por ejemplo, comentan el interés en sensar temperatura, vibraciones, sonido y radiación en los reactores para enviar los datos a la sala de control.

2.2.3. Eficiencia energética

Un campo donde se está explotando comercialmente es la eficiencia energética en el hogar y, sobre todo, en industrias. Empresas pioneras en redes de sensores enfocadas en productos de monitorización de energía, mediante el empleo de redes de sensores (algo mínimamente invasivo), pueden detectar patrones que permiten planificar mejor las cargas de trabajo en *data centers*.

Otro proyecto interesante ha sido EQUOS *Lighting System* de *Philips Lighting*, en el cual se utilizan balastos de lámparas fluorescentes equipadas con comunicación Zigbee para realizar tareas de mantenimiento, control y estudiar mecanismos de ahorro. Este proyecto ha culminado en un producto comercial llamado Hue [31], a medio camino entre las redes de sensores y el IoT.

2.2.4. Domótica y salud

En domótica el principal interés ha sido introducir una tecnología inalámbrica y ampliamente usada en redes de sensores en aplicaciones comerciales: Zigbee [32].

El proyecto *Eaton's Home Heartbeat System* iba un paso más allá buscando conectar todos los dispositivos de una casa. Estaba especialmente enfocado a seguridad, pero también permitía monitorizar y controlar diversos objetos. Por desgracia este proyecto no ha tenido continuidad.

Volviendo al terreno de la investigación, y en un campo a medio camino entre el hogar y la salud se sitúa el proyecto [33], donde se pretende aprovechar la infraestructura típica de un hogar para mejorar la calidad de vida de pacientes, la detección temprana de algunas dolencias y mejorar la comunicación entre el médico y el paciente. Por otro lado, en [34] se despliega una red que combina el sensado del ambiente con el de la salud de personas para determinar si es necesario activar un protocolo de asistencia.

2.2.5. Industriales y vehículos

En [35] se identifican varias áreas donde las redes de sensores pueden aportar un valor añadido, como son la detección de eventos, recolección de información, diagnóstico de máquinas y sistemas de seguridad. Empresas que utiliza redes de sensores para monitorizar el nivel de gas licuado de petróleo en tanques de almacenaje. O la NASA (*national aeronautics and space administration*), que emplea nodos para monitorizar remotamente pruebas de propulsión de cohetes.

En automoción ha habido varios fabricantes y consorcios que han presentado diversos proyectos para utilizar los conceptos de redes de sensores en coches. Por ejemplo, el programa V2V del proyecto Car2Car [36] pretende que los coches se comuniquen inalámbricamente entre sí para alertarse de peligros, obstáculos que puedan encontrarse en su camino y de la presencia de vehículos de emergencias.

2.3. Arquitectura

El desarrollo de una pila de protocolos fiable y eficiente en términos de energía es importante para soportar las múltiples aplicaciones en la WSNs. Dependiendo de la aplicación, dichas redes pueden consistir en decenas o cientos de nodos desplegados en el escenario bajo estudio como se muestra en la Figura 2.2. Cada nodo tiene la capacidad de recoger datos del entorno, de encaminarlos hasta la estación base, la cual a través de otras infraestructuras o redes hará posible la comunicación con el nodo administrador de tareas encargado de recibir las órdenes del usuario final.

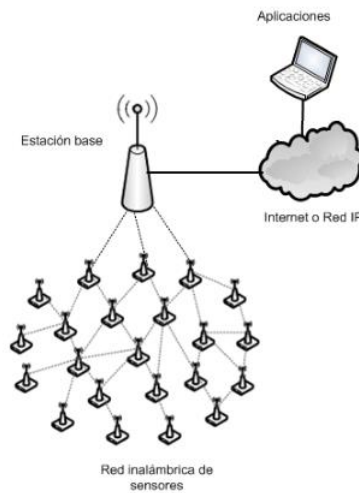


Figura 2.2: Infraestructura de una red inalámbrica de sensores

Cada nodo usa una pila de protocolos similares al modelo OSI (*open system interconnection*) para comunicarse con otros nodos: capa física, enlace de datos, red, transporte y aplicación.

La implementación de protocolos en las diferentes capas puede afectar significativamente al consumo de energía (principalmente las tres primeras capas), al retraso extremo a extremo y a la eficiencia del sistema. Es de vital importancia optimizar la comunicación y minimizar la energía empleada.

A continuación, se hace una introducción a los principales requerimientos de la capa física de una WSN, destacando los aspectos más influyentes en función del ahorro energético, pues en esta capa específicamente se basa el objeto de estudio del presente trabajo.

2.3.1. Capa física

La capa física proporciona un interfaz para la transmisión de flujos binarios sobre el medio físico. Es responsable de interactuar con la capa MAC (*media access control*), y proporciona capacidades de transmisión, recepción y modulación. Otras interacciones implican compartir información de canal y transmisión con la capa MAC para conseguir mayor rendimiento y mejor utilización de recursos.

Para una WSN, los problemas de reducir al mínimo el consumo de energía y maximizar el tiempo de vida de la red comienzan en la capa física. En esta capa, la energía se emplea en la operación del circuito de radio y en la transmisión. Mientras que la energía utilizada en la operación del circuito de radio es fija, la utilizada en el proceso de transmisión de datos puede variar según las pérdidas de paquetes, interferencias y distancia de transmisión. Existe un compromiso entre potencia de transmisión y la tasa de error producida. Una selección apropiada de la potencia de transmisión reduce al mínimo la pérdida de energía y hace operar a la red de forma más eficiente. Además, los esquemas de modulación son necesarios para transmitir datos sobre un canal inalámbrico. Así, se han desarrollado distintos esquemas de modulación para minimizar la probabilidad de error bajo diferentes condiciones [7].

Los esquemas de modulación energéticamente eficientes deberían reducir al mínimo tanto la energía de operación del circuito radio como la de transmisión. Recientes estudios de investigación incluyen como requerimientos de capa física un diseño simple de chips de radio de baja potencia y esquemas de transmisión y modulación [7].

La capa física se debe diseñar considerando los requerimientos de una WSN. Según las características de ésta, habrá restricciones en los *chips* de radio en términos de energía, tasas de datos, probabilidad de error, distancia de transmisión y fiabilidad.

Otras restricciones que deben considerarse en la capa física son las interferencias, la sincronización y la habilidad de transmitir en *multicast*. Si los nodos se despliegan de forma densa, la interferencia entre ellos puede ser inevitable. Cada nodo puede reducir su potencia de transmisión para reducir la interferencia; sin embargo, se necesita de sincronización entre los nodos. Debe haber sincronización entre la capa de enlace y la capa física, y entre los nodos. Con sincronización, la interferencia en la comunicación se puede reducir al mínimo.

Finalmente, los chips de radio con capacidad de *multicast* son útiles para transmitir datos a múltiples nodos al mismo tiempo. Así, todos los nodos necesarios recibirán la información con una sola transmisión.

2.3.1.1. Elección del ancho de banda

En WSNs, hay tres clases de tecnologías de capa física según el ancho de banda: banda estrecha, espectro ensanchado y ultra banda ancha (*ultra-wideband*). La tecnología de banda estrecha usa un ancho de banda del orden de la tasa de símbolo. Esta tecnología se centra en la eficiencia de ancho de banda, que es la medida de la tasa de datos sobre el ancho de banda utilizado. Por otra parte, en la tecnología de espectro ensanchado, la señal de banda estrecha es expandida. La función de ensanchado utilizada para determinar el ancho de banda es independiente del mensaje. Esta tecnología tiene la capacidad de reducir la potencia y permitir una comunicación con eficacia, esto es, con robustez ante interferencias y multitrayecto. Comparado con la tecnología de espectro ensanchado, *ultra-wideband* utiliza mayor ancho de banda, del orden del GHz. *Ultra-wideband* expande su señal sobre un ancho de banda tal que la interferencia producida a otros nodos es insignificante. Al igual que la tecnología de espectro ensanchado, *ultra-wideband* permite comunicaciones utilizando baja potencia de transmisión [7].

Varios estudios muestran que las tecnologías de espectro ensanchado cumplen los requerimientos de las WSNs mejor que las de banda estrecha. La tecnología de banda estrecha se centra en optimizar la eficiencia en ancho de banda y es menos robusta ante interferencias (excepto si se utiliza la técnica de *frequency hopping*), mientras que tanto la tecnología de espectro ensanchado como la de *ultra-wideband* combinan ancho de banda con ahorro energético.

Según el tipo de tecnología de espectro ensanchado, la sincronización puede ser beneficiosa debido a las propiedades de correlación de la secuencia pseudoaleatoria utilizada en el proceso de ensanchado. En cuanto a la capacidad *multicast*, los sistemas de banda estrecha no están diseñados para llevar a cabo esta tarea. En cambio, los sistemas de espectro ensanchado pueden realizar esta tarea con una asignación correcta de códigos pseudoaleatorios. *Ultra-wideband* tiene muchas características atractivas, pero comparado con tecnologías de espectro ensanchado, tiene sus propias restricciones y limitaciones [7].

2.3.1.2. Esquema de modulación

El esquema de modulación usado por el transceptor radio puede repercutir de forma sensible en el consumo de energía de un nodo. Son necesarios esquemas de modulación eficientes para minimizar el consumo energético.

La modulación multinivel transmite símbolos de un conjunto de M formas de onda diferentes, mientras que la modulación binaria usa únicamente dos formas de onda distintas. En general, la modulación multinivel requiere de circuitería más compleja y mayor energía. Sin embargo, está demostrado que la modulación M -aria es más eficiente energéticamente que la modulación binaria cuando el tiempo de encendido del transceptor es pequeño y la potencia de transmisión de radiofrecuencia es baja. En otra comparación, para un valor grande de M , la modulación M -FSK es más eficiente que la M -PSK y M -QAM, cuando M es mayor que ocho. Para valores menores de M , la modulación M -FSK no es tan eficiente porque se requiere de mayor potencia para conseguir la misma tasa de error de bit que en la modulación M -PSK y M -QAM. Sin embargo, para valores grandes de M , la relación señal a ruido requerida (SNR) en la modulación M -FSK crece lentamente, haciéndola muy eficiente en términos de energía [7].

En definitiva, la capa física en una WSN debe ser eficiente energéticamente. Dado que el diseño de la capa física comienza en el diseño del transceptor de radio, la elección de éste compromete el rendimiento del resto de capas de protocolo. El trabajo futuro se enfoca en nuevas innovaciones en el diseño de transceptores radio de bajo consumo con tecnologías emergentes, explorando las técnicas de *ultra-wideband* como alternativa para la comunicación, creando esquemas de modulación simples para reducir la sincronización y el coste de energía, buscando siempre la potencia de transmisión óptima.

2.4. Estándares

El valor de las redes inalámbricas de sensores se basa en su bajo coste y su distribución en grandes cantidades. Para lograr las economías de escala necesarias para alcanzar un mercado de bajo coste, algunos elementos de las redes inalámbricas de sensores deben ser estandarizados, de manera que los productos de muchos fabricantes puedan interoperar.

En la actualidad no existe un estándar de facto, ni en protocolos, ni en hardware, ni en representación de datos. Esto se debe a que las aplicaciones de las redes de sensores son tan amplias que es difícil englobarlas todas, así como que las tecnologías están en constante cambio y desarrollo. De entre las soluciones existentes, sin embargo, existen algunas con un amplio grado de utilización por varios fabricantes. A continuación, se presentan las principales iniciativas en curso.

2.4.1. Estándar IEEE 802.15.4

El estándar IEEE 802.15.4, finalizado en el 2003 por el Instituto de Ingenieros Eléctricos y Electrónicos, define la capa física y MAC para redes inalámbricas de área personal (WPAN) de baja tasa de transmisión. A veces se confunde el estándar con ZigBee, otro estándar que agrega servicios de red, seguridad y aplicación, y está basado en los servicios ofrecidos por IEEE 802.15.4. Los tipos de aplicación a los que está orientado comprenden las redes inalámbricas de sensores, la domótica, las redes hogareñas, la conexión de dispositivos a una computadora personal, seguridad, etc. La mayoría de estas aplicaciones requieren tasas de transmisión bajas a medias, retardos de transmisión moderados con requerimientos no muy estrictos, y es muy deseable la reducción al mínimo del consumo de energía en los nodos [37].

El diseño de la capa física está dirigido por los requerimientos de bajo costo y eficiencia, de aplicaciones de control y monitoreo sensibles al costo y de baja tasa de transmisión. Bajo el estándar 802.15.4, se pueden operar enlaces inalámbricos en tres bandas de frecuencias no licenciadas:

1. La banda de los 868.0-868.6 MHz para Europa, que permitía un solo canal de comunicación en el estándar de 2003. Ha sido aumentado a 3 canales en la versión del estándar de 2006.
2. En Norte América opera en la banda de los 902-928 MHz. En el primer estándar de 2003, esta permitía hasta 10 canales de comunicación. En el estándar de 2006 el número de canales aumentó a 30.

3. En general, en todo mundo, incluyendo Europa y Norte América, opera en la banda de los 2400-2483.5 MHz, donde puede tener hasta 16 canales (estándares 2003 y 2006). Esta banda es la más utilizada.

Basados en estas frecuencias, se definen tres medios físicos:

1. DSSS (*direct sequence spread spectrum*) usando modulación BPSK en la banda de 868 MHz a una tasa de 20 Kbps (único canal).
2. DSSS usando modulación BPSK en la banda de 915 MHz a una tasa de 40 Kbps (10 canales).
3. DSSS usando modulación O-QPSK (*offset quadrature phase-shift keying*) en la banda 2.4 GHz a una tasa de 250 Kbps (16 canales)

El estándar IEEE 802.15.4-2007 es una enmienda que especifica las siguientes alternativas adicionales de capa física [38]:

1. *Ultra-wide band* (UWB) en las frecuencias 3 a 5 GHz, 6 a 10 GHz, y menores 1GHz.
2. CSS (*chirp spread spectrum*) a 2450 MHz.

En abril de 2009 se publicaron dos revisiones más del estándar, IEEE 802.15.4c [39] e IEEE 802.15.4d [40], que aumentaban los tipos de capas físicas: una capa en los 780 MHz, con codificación O-QPSK o MPSK. Otra capa en la banda de los 950 MHz, que utiliza GFSK o BPSK.

En la tabla 2.1 se muestran todos los tipos de capas físicas contempladas en el estándar IEEE 802.15.4.

Tabla 2.1: Resumen capa física estándar 802.15.4.

Modo propagación ²	Banda frecuencias	Modulación	Publicación
DSSS	868.0-868.6 MHz	BPSK	802.15.4-2006
DSSS	902.0-928.0 MHz	BPSK	802.15.4-2006
DSSS	2400-2483.5 MHz	O-QPSK	802.15.4-2006
DSSS UWB	< 1 GHz, 3-5 GHz, 6-10 GHz	Sin especificar	802.15.4a-2007
CSS	2450 MHz	DPSK	802.15.4a-2007
DSSS	780 MHz	O-QPSK, MPSK	802.15.4c-2009
DSSS	950 MHz	GFSK, BPSK	802.15.4d-2009

La estructura de la trama IEEE 802.15.4, ilustrada en la figura 2.3, comprende los siguientes campos [41]:

1. Preámbulo: 32 bits que se utilizan para sincronización de símbolos.
2. Delimitador: 8 bits que se utilizan para sincronizar la recepción de la trama.
3. Cabecera: 8 bits que especifican la longitud de la unidad de datos (PSDU, *physical service data unit*).
4. Datos: hasta 127 bytes de datos.

² Un error muy cometido, es la confusión entre las técnicas de propagación y modulación. La diferencia radica en: que la técnica de propagación distribuye la información a través de una variedad de canales, en tanto la técnica de modulación modula la información a través de cada uno de esos canales.

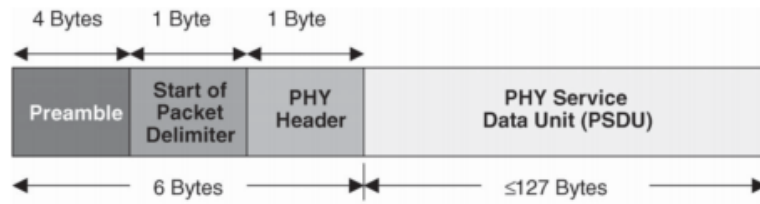


Figura 2.3: Estructura de la trama de capa física IEEE 802.15.4 [41].

La capa física del protocolo realiza las funciones de: activación y desactivación del transceptor de radio, detección de energía del canal actual, indicador de calidad de enlace para paquetes recibidos, evaluación de borrado de canal para la detección de portadora de acceso múltiple con el mecanismo de prevención de colisiones CSMA-CA (*carrier sense multiple access with collision avoidance*), selección de la frecuencia de canal y transmisión y recepción de datos.

La capa MAC controla el acceso al canal radio usando el mecanismo CSMA-CA. La coexistencia de otros servicios que usan bandas sin licencia con los dispositivos IEEE 802.15.4 fue también un factor importante en el diseño del protocolo, y es evidente en muchas de sus características. Por ejemplo, es necesaria la selección dinámica de canales; cuando se puede interferir a otros servicios que se encuentran trabajando en un canal que está siendo utilizado por una red IEEE 802.15.4, el nodo de red que controla la misma (PAN, coordinador de red de área personal) explora otros canales disponibles para encontrar un canal más adecuado. En esta exploración, se obtiene una medida de la energía de cada canal alternativo y, a continuación, utiliza esta información para seleccionar un canal adecuado. Este tipo de escaneo también se puede usar antes del establecimiento de una nueva red. Antes de cada una de las transmisiones (excepto tramas baliza o de reconocimiento), cada nodo de red IEEE 802.15.4 debe realizar dos evaluaciones de canal (CCA, *clear channel assessment*) como parte del mecanismo CSMA-CA para garantizar que el canal está desocupado antes de la transmisión [41].

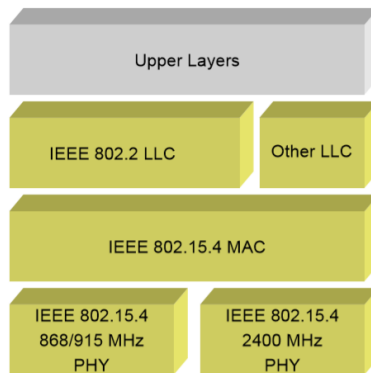


Figura 2.4: Protocolos de IEEE 802.15.4 [41].

2.4.2. ZigBee

ZigBee [17] es un estándar de comunicación inalámbrica orientado a aplicaciones cuyos requerimientos principales son bajas tasas de transmisión, fácil aplicación, alta fiabilidad, bajo costo y larga duración de batería.

Los estándares ZigBee son desarrollados por la organización ZigBee Alliance, conformada por cientos de compañías, y formada en el 2002 como una organización sin fines de lucro.

El estándar ZigBee está definido en capas de protocolos, basadas en el modelo de referencia OSI. Adoptando la capa física y de acceso al medio IEEE 802.15.4, define capas de servicios de red, aplicación y seguridad.

Los dispositivos ZigBee pueden formar redes con topología en malla, estrella y árbol, que conectan desde cientos a unos miles de dispositivos que no exijan una gran cantidad de transmisión de datos. Además, éstos requieren muy poca energía y pueden funcionar con pequeñas baterías durante muchos años. En una red ZigBee pueden participar dos tipos diferentes de dispositivos: dispositivos de funcionalidad completa (FFD, *full function device*) y dispositivos de funcionalidad reducida (RFD, *reduced function device*). Los FFD pueden operar en tres

modos distintos: coordinador (*Zigbee coordinator*), encaminador (*Zigbee router*) y el dispositivo final (*Zigbee end device*). El coordinador es el que inicia la formación de red, almacena la información y puede hacer de puente entre redes. El encaminador une grupos de dispositivos y proporciona comunicación salto a salto (*multi-hop*) a través de los dispositivos. Los dispositivos finales consisten en sensores, actuadores y controladores que recogen datos y se comunican sólo con el encaminador o el coordinador.

En la topología en estrella, la comunicación se establece entre los dispositivos y un controlador central único, denominado coordinador de la WPAN (*wireless personal area network*). En la topología de malla, cualquier dispositivo puede comunicarse con cualquier otro dispositivo, siempre y cuando estén uno en el rango del otro. La red en árbol es un caso especial de una red en malla en la que la mayoría de los dispositivos son FFD y un RFD puede conectarse a la red como un nodo hoja en el extremo de una rama. Cualquiera de los FFD puede actuar como *router* y proporcionar servicios de sincronización a otros dispositivos y *routers*. El coordinador de la WPAN será uno de estos routers. Ver figura 2.5.

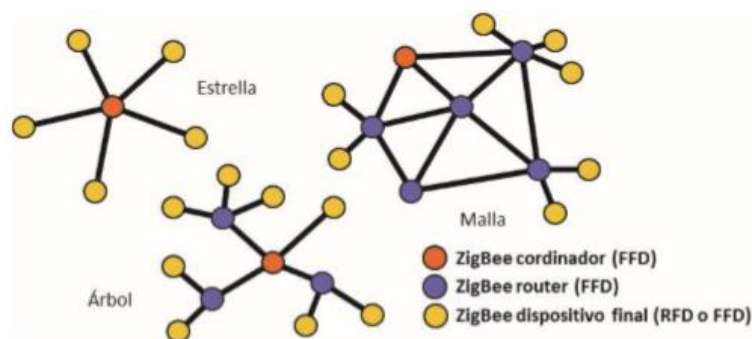


Figure 2.5: Diagrama de la estructura de una red ZigBee [19].

2.4.3. Otros estándares

Además de los conocidos estándares IEEE 802.15.4 y Zigbee, se han desarrollado muchos otros, aunque para aplicaciones más específicas como es el caso de WirelessHART (*wireless highway addressable remote transducer*).

WirelessHART surge como una adaptación de la especificación HART (*highway addressable remote transducer*) ya existente, que se creó en Estados Unidos para la monitorización de sensores en entornos industriales. Esta iniciativa especifica perfiles y casos prácticos en los que se puede aplicar directamente el control inalámbrico [42]. Conceptualmente, las redes WirelessHART son un tipo especial de redes inalámbricas de sensores. A diferencia de las redes de sensores genéricas, que asumen que los sensores son desplegados al azar y de manera abundante, el despliegue de las redes WirelessHART es preciso y con redundancia limitada. Los nodos WirelessHART están conectados a dispositivos de campo para recolectar datos ambientales específicos de los procesos, por ejemplo, velocidades de flujo, niveles de fluido, o temperaturas. Una lectura de un sensor quizá no puede reemplazarse por la lectura de otro sensor cercano. La capa física está mayormente basada en la capa física DSSS 2.4 GHz definida en el estándar IEEE 802.15.4 de 2006, operando en la banda ISM (*industrial, scientific and medical*) de uso no licenciado 2400-2483.5 MHz, con una tasa de transmisión de 250 Kbits/s y 16 canales con 5 MHz de separación entre cada canal adyacente y bajo una modulación O-QPSK. Su *payload* máximo es de 127 bytes. La capa de enlace es propia y opera de manera sincronizada, con ranuras estrictas de 10 ms y tecnología TDMA (*time division multiplexed access*) para proveer un sistema de comunicación determinístico y libre de colisiones.

Otro protocolo interesante resulta 6LoWPAN [43] (*IPv6 over low power wireless personal area networks*) que permite la comunicación de paquetes IPv6 sobre una red basada en el estándar IEEE 802.15.4. Usando 6LoWPAN, los dispositivos de bajo coste tienen todas las ventajas de comunicación y gestión IP. El estándar proporciona una capa de adaptación, nuevo formato de paquete y gestión de direccionamiento. Como el tamaño de paquete IPv6 es mucho más grande que el tamaño de la trama IEEE 802.15.4 es necesaria una capa de adaptación. La capa de adaptación lleva a cabo la funcionalidad de compresión de cabeceras, en ella se crean

paquetes de menor tamaño que se integran en una trama IEEE 802.15.4. Soporta velocidades de 250 kbps, 40 kbps, y 20 kbps según lo definido en cada una de las capas físicas.

Como se puede apreciar existen una infinidad de estándares de comunicación aplicables a redes de sensores, la mayoría de ellos utilizan el estándar IEEE 802.15.4 como base de su capa física y de acceso al medio con algunas modificaciones. Todos cumpliendo con el requerimiento fundamental en cuanto a la eficiencia energética.

2.5. Hardware

Dada la gran variabilidad de aplicaciones de las WSNs, existen múltiples fabricantes que desarrollan plataformas de *hardware*. Las plataformas descritas a continuación implementan la capa física del estándar IEEE 802.15.4, poseen capacidades de procesamiento y memoria suficientes para poder implementar ZigBee y ser destinadas a aplicaciones de sensado remoto en WSN. Están preparadas para trabajar en modo de bajo consumo y son de bajo costo.

Principales fabricantes de plataformas de *hardware*:

- **NXP:** *NXP Semiconductor Inc. (Next eXPerience)*, es una empresa fabricante de semiconductores que se creó el 31 de agosto de 2006 a partir de la división de semiconductores de la empresa holandesa *Philips*. Son varios los modelos NXP ha fabricado, diferenciándose principalmente en la parte de radiofrecuencia. El modelo más básico es el JN5139-xxx-M00. Dispone de una antena integrada con la que se consigue un rango de transmisión de 1km, la misma distancia que podemos conseguir con los modelos JN5139-xxx-M01/M03 que incorporan un conector SMA (*subminiature version A*), pero no tienen amplificador. Los modelos más avanzados son los JN5139- xxx-M02/M04 que incorporan un conector SMA junto con un amplificador de potencia y un amplificador de bajo ruido para incrementar el rango de transmisión hasta los 4 Km [44][45]. NXP adquirió en 2015 a *Freescale Semiconductor Inc.* otra de las empresas dedicadas al diseño de módulos para WSN.
- **Digi:** *Digi International Inc.* Produce varios dispositivos de interés para el diseño de WSN, como es el caso de los módulos Xbee ZB y Xbee-PRO ZB, ambos implementando el protocolo ZigBee. La diferencia entre ellos es la potencia de transmisión y por tanto el alcance. Su uso es muy simple por lo cual se los utiliza frecuentemente con fines educativos. Son dispositivos que no necesitan ser programados para aplicaciones simples (como transmitir datos sin restricciones específicas como bajo consumo), pero será necesario un estudio y configuración muy cuidadoso para utilizarlos en una WSN. Poseen un chip Ember EM250 el cual incluye un microcontrolador dedicado y un tranceptor 802.15.4. Debido a su firmware, solo pueden configurarse algunos de sus parámetros, por lo que para aplicaciones más complejas será necesario comunicarse con un microcontrolador de uso general a través de una interfaz serie. Los módulos Xbee de Digi poseen la certificación de la *Zigbee Alliance*, lo que garantiza que estos módulos puedan comunicarse con cualquier plataforma certificada por la *Zigbee Alliance* [46].
- **Microchip:** *Microchip Technology Inc.* Empresa fabricante de microcontroladores, memorias y semiconductores analógicos. Luego de la adquisición de *Atmel Corp.* en 2016 paso a comercializar sus productos, enfocados en módulos para redes inalámbricas bajo el estándar Zigbee.
- **Silicon Labs:** *Silicon Laboratories Inc.* es un proveedor líder de soluciones de sensor inteligente se caracteriza por excelente confiabilidad, tamaño compacto, altos niveles de integración, inigualable facilidad de uso para una variedad de aplicaciones. Incluye sensores ópticos, digital de humedad relativa, de temperatura, biométricos, magnético. Adquiere la compañía Telegesis, que ya utilizaba tecnología ZigBee de *Silicon Labs* en sus módulos de productos: desplegados en los contadores inteligentes, adaptadores USB y las entradas para aplicaciones de energía inteligente. Aplicaciones adicionales incluyen la automatización del hogar, conexiones de iluminación, seguridad y automatización industrial. Telegesis también ofrece kits de evaluación para ayudar a los desarrolladores a optimizar sus aplicaciones basadas en ZigBee y desarrollo integral.

La Tabla 2.2 muestra parámetros significativos para evaluar la aptitud de una plataforma a utilizarse como nodo en una WSN.

Tabla 2.2: Resumen de datos de algunas plataformas disponibles en el mercado para su empleo en WSN.

Fabricante	Modelo	Pot. Máx. (mW)	Tensión de alimentación (V)	Protocolo embebido	Consumo			Sensibilidad (dBm)
					Modo sleep (μ A)	Modo Tx (mA)	Modo Rx (mA)	
Digi	Xbee ZB	2	2,1 a 3,6	ZigBee	<1	38	37	-96
Digi	Xbee-PRO ZB (S2B)	10	2,7 a 3,6	ZigBee	4	205	47	-102
NXP	JN5139-xxx-M00/01/03	1.8	2,7 a 3,6	IEEE 802.15.4	2.6	37	37	-96
NXP	JN5139-xxx-M02/04	80	2,7 a 3,6	IEEE 802.15.4	2.6	125	45	-100
Digi	Xbee-PRO ZB (S2B)	63	2,7 a 3,6	ZigBee	3,5	220	62	-102
Microchip	MRF24J40MA	1	2,4 a 3,6	IEEE 802.15.4	2	23	19	-94
Telegesis ³	ETRX357	2	2.1 a 3.6	ZigBee	2.2	31	25	-99

No todos los fabricantes indican el alcance del enlace de sus dispositivos. El alcance está en estrecha relación con la potencia del transmisor y la sensibilidad del receptor. Por ejemplo, Digi declara un alcance de 120 metros para sus nodos Xbee ZB (potencia de transmisión 2 mW y sensibilidad -95 dbm, uso *indoor*) y un mínimo de 1500 metros para sus nodos Xbee Pro ZB de uso *outdoor* (potencia de transmisión 10 mW y sensibilidad -102 dbm). *Microchip* especifica un alcance de 122 metros para sus nodos MRF24J40MA (potencia de transmisión 1 mW y sensibilidad -94 dbm). Tampoco todos los fabricantes brindan los valores máximos y mínimos de corriente consumida en modo sleep (bajo consumo), transmisión y recepción, en la comparativa se han indicado los valores típicos (dato disponible en todos los *datasheet*).

2.5.1. Motes con protocolo ZigBee

Debido a que los estándares más populares para redes de sensores inalámbricos son IEEE 802.15.4 y ZigBee, la mayoría de sus fabricantes incluyen estos estándares en la operatividad de sus productos.

Actualmente podemos encontrar módulos Zigbee de la mano de fabricantes como *Digi International*, *Telegesis* o *NXP* que otorgan ciertas facilidades en el desarrollo de aplicaciones, proporcionando una versión del *stack* Zigbee certificado, aunque no todos los fabricantes ofrecen la base de software de forma gratuita. Generalmente estos módulos tienen un coste relativamente bajo si lo comparamos con otras tecnologías inalámbricas dependiendo del nivel de desarrollo de este.

La manera más básica de implementar un módulo Zigbee es mediante un microcontrolador comercial. Esta es la opción que más tiempo requiere teniendo que desarrollar casi toda la parte hardware, incluyendo la parte de radiofrecuencia que suele ser la más problemática. Por otra parte, una manera más eficaz, bastante más rápida, pero más cara de implementar, es utilizar uno de los módulos existentes en el mercado con el hardware ya desarrollado, que facilita la programación/depuración del microcontrolador, incluyendo conectores e integrados de utilidad como pueden ser sensores, una memoria EEPROM (*electrically erasable programmable read-only memory*), un depurador embebido aparte de diodos LED y botones de programación software o inicialización [47].

A diferencia de las opciones anteriores, los módulos más comunes incorporan solamente el microcontrolador junto con un bloque de radiofrecuencia formado por líneas de transmisión y una o varias antenas. A continuación,

³ Adquirida por *Silicon Laboratories Inc.* Aunque los productos mantienen el nombre de *Telegesis*.

se realizará una descripción de los módulos ZigBee que actualmente están en el mercado y que son utilizados por las empresas para desarrollar sus aplicaciones indicando sus principales características.

2.5.1.1. Xbee



Figure 2.6: Módulo ZigBee Xbee.

Hay diferentes modelos de este módulo de transmisión por RF fabricado por la empresa *Digi International*. Cada uno de ellos tiene diferentes características en cuanto a la topología, el rango de transmisión y demás características de radio. De todos ellos el que ofrece mejores prestaciones es el modelo “Programable XBee-PRO ZB” que incluye un microcontrolador con arquitectura HCS08 con 32 KB Flash / 2 KB RAM (*random access memory*) de memoria dedicadas al programa de aplicación del usuario y que permite la creación de redes ZigBee en retícula o malla (Apéndice A.4).

Estos dispositivos están configurados y programados de fábrica para establecer comunicación sin necesidad de que el usuario haga prácticamente nada, es decir, estos dispositivos se encargarían en un nodo de una red ZigBee de la comunicación, por lo que deben conectarse por puerto serie con otro microcontrolador que se encargue del procesamiento de los datos sensados o recibidos.

Es posible conectar una red XBee una red más amplia tipo Ethernet, por medio de los *gateways* que proporciona *Digi Internacional* llamados XBee Gateway. Estos dispositivos proporcionan una interfaz entre la red ZigBee y una red IP a un número ilimitado de nodos XBee, así como permiten configurar y gestionar una red XBee. Están programados en el lenguaje de programación *Python*, existiendo un entorno de desarrollo de código abierto para facilitar el desarrollo de la aplicación [47].

2.5.1.2. ETRX357



Figure 2.7: Módulo ZigBee ETRX357.

Las series ETRX3 de Telegesis son módulos ZigBee a 2.4 GHz de baja potencia que integran un transceptor compatible con IEEE 802.15.4. Utiliza una arquitectura eficiente que supera los requisitos de rango dinámico impuestos por el estándar IEEE 802.15.4 en más de 15 dB. El filtrado de canal integrado permite una robusta coexistencia con otros estándares de comunicación en el espectro de 2,4 GHz, como IEEE 802.11 y Bluetooth.

Para mantener los requisitos de tiempo estrictos impuestos por el estándar 802.15.4 el ETRX3 integra una serie de funciones de control de acceso al medio en el manejo de la transmisión y recepción automática de ACK (*acknowledgement*), retardo de reenvío automático, evaluación del canal para la transmisión, así como filtrado automático de los paquetes recibidos [47].

2.5.1.3. JN5139-001-MXX



Figure 2.8: Módulos ZigBee JN5139-001-MXX.

Este módulo de la compañía NXP es uno de los más interesantes debido a los distintos modelos que podemos encontrar según los requerimientos de la aplicación y a las prestaciones del microcontrolador que incorpora.

Las características más destacables del módulo JN5139-001-M02/04 son su elevada potencia de transmisión, lo que permitiría implementar aplicaciones en las que sea necesario cubrir una zona extensa, reduciendo los costes. Además, el enlace tendrá mejor calidad si los nodos son colocados a una distancia intermedia relativa al máximo alcance (4 Km de rango), siendo esta una distancia muy considerable para un enlace ZigBee y la arquitectura de 32 bits del microcontrolador junto con la capacidad de memoria. Los 192 KB de memoria flash permite la integración de los *stacks* de ZigBee tanto para una topología punto a punto como mallada y los 96 KB de RAM permiten integrar funcionalidades de *router* y de control, ya que estas aplicaciones necesitan utilizar una gran cantidad de variables que deben ser almacenadas en RAM [47].

2.6. Optimización del consumo energético en redes de sensores inalámbricos

Un nodo sensor, siendo un dispositivo electrónico de pequeñas dimensiones, sólo puede estar equipado con una fuente de alimentación limitada. En algunas aplicaciones, la sustitución o recargas de baterías puede tornarse imposible, por lo que el tiempo de vida útil de un nodo presenta una fuerte dependencia con la duración de su batería.

En otras redes, la potencia consumida ha sido un factor de diseño importante, pero no fundamental, simplemente porque los recursos energéticos pueden ser sustituidos por el usuario y, por lo tanto, el diseño se centra más en la calidad de servicio proporcionada que en la eficiencia energética. En WSNs, sin embargo, la eficacia energética es una característica fundamental que influye directamente en el tiempo de vida de la red.

La tarea principal de un nodo desplegado sobre el entorno bajo estudio es detectar eventos, realizar localmente un procesamiento rápido de datos y luego transmitirlos. Por tanto, el consumo de energía se puede dividir en tres dominios: sensado, procesamiento de datos y comunicación. Ver figura 2.9.

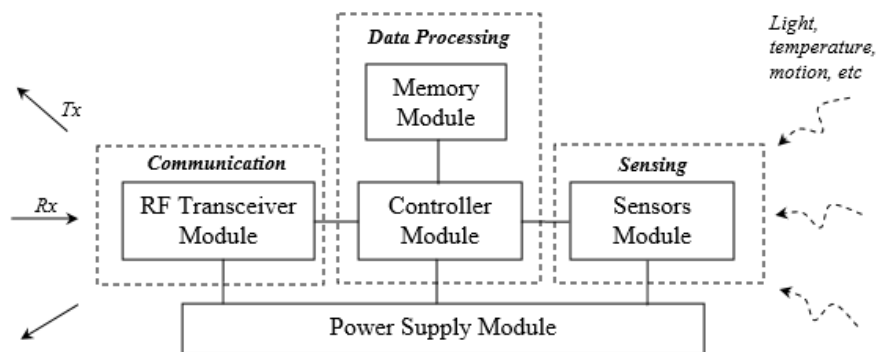


Figura 2.9: Dominios del consumo energético en WSN [48].

La energía utilizada en el sensado varía según la naturaleza de la aplicación. Así, las medidas esporádicas consumirán menos energía que una monitorización continua. Además, niveles altos de ruido podrían causar una degradación significativa y aumentar la complejidad del algoritmo de detección, provocando un mayor gasto energético. De los tres dominios, un nodo sensor gasta más energía en el proceso de comunicación de datos. Esto

implica tanto transmisión como recepción. De ahí, que sea crucial el procesado local, utilizando alguna estrategia que reduzca al máximo el número de bits a modular y por ende a ser transmitidos [7].

Muchos son los trabajos relacionados en la búsqueda de optimizar los recursos energéticos en las WSNs, desde la línea de investigación conocida como *energy harvesting* o *energy scavenging*, basada en la necesidad de que los propios nodos sensores consigan la energía con la cual autoabastecerse.

En este aspecto una de las fuentes de energía más utilizadas es la solar. Esto se debe a que tanto la fuente de energía como la tecnología están muy extendidas y se obtienen rendimientos muy altos comparados con otras fuentes. Por ejemplo, en [49] se propone un seguidor del punto de máxima potencia de paneles solares aplicado a nodos de redes de sensores para obtener la máxima energía con la que recargar una batería; para ello emplean un convertidor DC/DC variable controlado por un microcontrolador.

Otra fuente que se ha aplicado ampliamente es la piezoeléctrica, donde se extrae energía de movimientos o vibraciones. Se ha aplicado en suelos para extraer energía de las pisadas, en ropa para extraer energía de los movimientos de personas o por ejemplo en [50], donde se utilizan celdas de material piezoeléctrico para aprovechar las corrientes de agua en mares y ríos para generar unos pocos milivatios, pero suficientes para cargar una batería.

Además, también ha habido intentos de extraer energía de las señales de RF en el ambiente. En [51] se ideó un sistema analógico para aprovechar la energía de emisiones de radio y cargar un condensador con el que poder alimentar un pequeño nodo.

En el terreno del *hardware* se pueden comentar algunos puntos evidentes, como apagar todos los dispositivos que no se utilicen, tanto externos, como periféricos internos del microcontrolador. Este hecho tan trivial impone que sea conveniente utilizar dispositivos que permitan manejar modos de bajo consumo y tengan unos tiempos de activación cortos. En el caso de dispositivos que no tienen esos modos de bajo consumo, por ejemplo, sensores analógicos, conviene desarrollarlos, bien mediante interruptores en su alimentación, alimentándolos desde líneas digitales del microcontrolador (si aportan corriente suficiente) u otros [52]. También es recomendable ajustar los valores de las típicas resistencias de *pull-up* de los *bus* de comunicación lo más alto posible para la velocidad de transmisión deseada, dado que presentan un consumo cuando la línea está a nivel bajo [53].

En general los voltajes de trabajo menores presentan también unos consumos menores, por lo que es conveniente alimentar el circuito a la tensión más baja posible, o si está disponible la opción, utilizar una tensión baja únicamente para el núcleo del procesador. En [54] proponen un procesador con PMU (*power management unit*), el cual responde a estados y va cambiando de modo, además de variar la tensión de alimentación a través de un convertidor DC/DC variable.

La mayor parte de las técnicas anteriores están dirigidas a los microcontroladores, ya que son los que controlan este tipo de nodos. Entre los microcontroladores que mejor manejan el consumo destacan *Texas Instruments* con los MSP430 y los nanoWatt de *Microchip*.

El otro gran componente común a todos los motes que tiene mayor efecto en el consumo es el *transceiver*. Además de tener diferentes modos de funcionamiento como los microcontroladores, los *transceivers* generalmente permiten regular su potencia de transmisión, por lo que ésta puede ajustarse dinámicamente para reducir el consumo. También se han realizado estudios sobre los mejores esquemas de modulación, etapas de potencia más eficientes, etc. La forma más efectiva de ahorrar energía en los *transceivers* es llevarlos a modos de bajo consumo cuando no son necesario, si los protocolos de comunicaciones lo permiten [55]. No obstante, para mantener la comunicación, los diferentes nodos deben establecer un mecanismo de sincronización para que unos envíen datos cuando otros escuchan. Así, desde el *firmware* se establecerán mecanismos como los esquemas de ciclos de trabajo dormido/desperto o equivalentes para reducir su consumo.

Desde el punto de vista del *firmware* una técnica que a veces se utiliza es tener un manejo variable de la frecuencia de reloj. Dado que los nodos de las redes de sensores suelen tener ciclos de trabajo periódicos, sería posible utilizar un reloj de una determinada frecuencia en algunos estados (por ejemplo, en estado dormido un reloj RC de baja frecuencia) y de otra frecuencia diferente en otros estados (por ejemplo, en estado *listen* un reloj de cristal de mayor frecuencia).

Por otro lado, también se ha medido que energéticamente cuesta más transmitir un byte que ejecutar cien instrucciones [55][56], por tanto, puede ser conveniente establecer estrategias para reducir la cantidad de bytes a

enviar, tanto de la propia información de los sensores como del sobrecoste que supone la gestión de las comunicaciones y la retransmisión en caso de errores.

Cuando se habla de extender la vida de los nodos sensores, normalmente se habla de reducir su consumo medio para aumentar el tiempo que una carga inicial de la batería lo puede mantener alimentado. No obstante, en nodos que puedan autoabastecerse y, por tanto, tengan mecanismos de recarga de la batería como, por ejemplo, paneles solares, aparece otro factor que puede ser limitante: el desgaste de las baterías. Si hay un gran desgaste en las baterías, será necesario un mantenimiento más frecuente para comprobarlas y/o cambiarlas. Relacionado con este aspecto, en [57] apuntan a técnicas para reducir este desgaste y, por tanto, extender la vida de las baterías en las redes de sensores como por ejemplo que la demanda de corriente por parte de la carga conectada a la batería sea una corriente continua y baja, para lo que proponen aumentar el ciclo de trabajo a cambio de bajar la tensión de alimentación.

Los protocolos son uno de los puntos más tratados por los investigadores en la búsqueda de mejorar la eficiencia energética. Como se ha comentado antes, los tres primeros niveles de la pila de protocolos tienen un impacto fundamental en el consumo de energía.

En cuanto al nivel físico y de acceso al medio se puede comentar una de las primeras propuestas: SMAC [58]. Este protocolo es sencillo: utiliza un esquema despierto/dormido donde el período despierto se subdivide en tres ranuras temporales (*slots*): SYNC, RTS (*request to send*) y CTS (*clear to send*). En el primer slot se envía un paquete de sincronización, el cual es utilizado por los nodos que lo reciben para ajustar su base de tiempos respecto al emisor; en RTS los nodos hacen una petición de envío de datos en caso de que necesiten transmitir, e indican a quién enviarán esos datos; en CTS el nodo coordinador confirma el envío al nodo que lo solicitó. En el siguiente período los nodos, en caso de no tener que enviar o recibir datos, podrán pasar al modo dormido; mientras que, si están involucrados en la transmisión, permanecerán en el estado adecuado hasta que finalice el envío y éste se confirme con un ACK (paquete de confirmación de llegada de datos).

Sobre este protocolo se han propuesto algunas modificaciones, como DSMAC, el cual presenta un ciclo de trabajo variable o T-MAC (*timeout MAC*) [59], que juega con las ventanas de RTS y CTS.

Otra propuesta es BMAC [60]. Este protocolo supone que hay un nodo central con alimentación perpetua. Los nodos finales se despertarán periódicamente (intervalo de chequeo) para comprobar brevemente si hay actividad en el canal; si la hay permanecerán a la escucha; si no, pasarán al estado dormido. El nodo central emitirá un preámbulo de duración mayor al intervalo de chequeo para que se sincronicen con él los nodos con los que haya que establecer comunicación y, al finalizar, se enviarán los datos.

Una modificación de este protocolo es WiseMAC [61] que combina el mecanismo tradicional de CSMA/CA con el BMAC, pero éste último ajustará la duración de su preámbulo dinámicamente para reducir los tiempos despiertos.

La topología afecta directamente a la forma en que se transmiten los datos, principalmente en el direccionamiento. La forma de comunicación más directa es punto a punto, en la que hay alcance directo entre el nodo emisor y el receptor; no obstante, esta circunstancia no puede darse en todos los casos porque el rango de cobertura limita la dispersión geográfica, porque la aplicación requiere segmentar la red, porque los nodos son móviles y reubicables. Por tanto, en muchos casos será necesario encaminar la información de un nodo a otro de forma que ésta puede pasar por diversos nodos intermedios; así se puede hablar de redes multisalto (*multihop*).

El rendimiento de las WSNs está estrechamente relacionado con el protocolo de enrutamiento, debido a que las rutas pueden variar dinámicamente en el tiempo. Sino existe un protocolo de enrutamiento eficiente, se pueden llegar a consumir una cantidad significativa de energía debido al intercambio intensivo de mensajes. Diferentes protocolos de enrutamiento para redes de sensores inalámbricos han sido propuestos en la literatura. Uno de los más populares debido a su buen rendimiento en escenarios de alta movilidad, aumento en la densidad de la red y cargas de poco tráfico, es AODV (*ad-hoc on demand distance vector*). Este es implementado por el estándar ZigBee en su capa de red.

En [62] se propone DH-AODV (*directional hierarchical AODV*) que aprovechando los campos ya existentes en los paquetes de control del protocolo AODV, reduce el número de paquetes de descubrimiento de ruta (RREQ) que inundan la red, así como el número de paquetes perdidos, lo que se traduce en menores tiempos de actividad de los nodos y con ello un ahorro energético.

En conclusión, muchas y diversas han sido las propuestas en cuanto a control y gestión de la energía en las WSN: desde el punto de vista de *hardware*, *software*, protocolos, etc. Sin embargo, poco o nada se ha desarrollado en cuanto a cambios en las modulaciones que busque disminuir el gasto energético en la transmisión de datos entre los motes.

La capa física de un sistema de comunicaciones es la encargada de llevar a cabo funciones y servicios relacionados con la estandarización de los medios físicos (especificaciones mecánicas y eléctricas), codificación, modulación, sincronización, definición de topologías, entre otras funciones. Uno de los procesos más importantes de esta capa, es el proceso de codificación y modulación.

En este capítulo se ofrece una descripción de las bases teóricas de las modulaciones digitales; se caracterizan las modulaciones tomadas como base para el desarrollo de la nueva técnica de codificación-modulación: BPSK y OOK (caso especial de la modulación ASK). Finalmente se presenta la técnica de codificación-modulación BPSK^{+S} con una descripción detallada de su funcionamiento.

3.1. Modulación

En telecomunicaciones, la modulación es un proceso que consiste en modificar las características de una señal (amplitud, frecuencia o fase) llamada portadora, en función de otra señal que contiene la información que se desea transmitir, llamada moduladora, con el objetivo de tener una nueva señal (señal moduladora) más adecuada al medio de transmisión (guiado o no guiado) y a las características del mismo: ancho de banda, atenuación, impedancia característica, interferencia electromagnética, entre otras.

Los sistemas de modulación suelen clasificarse en analógicos y digitales. La modulación analógica emplea como moduladora una señal continua, la modulación digital, en cambio, es un proceso mediante el cual se transforman símbolos digitales en formas de onda adecuadas para la transmisión en un canal de comunicación.

Existen diversas ventajas que surgen de la modulación, entre las principales encontramos [63]:

- **Empleo eficiente del espectro radioeléctrico:** Al modular las señales, se permite colocarlas en diferentes bandas de frecuencia del espectro, lo que permite administrarlo.
- **Transmisión múltiple:** La transmisión múltiple, permitida por la modulación, puede trasladar a diferentes posiciones en el dominio de la frecuencia las señales de información. Por lo que los espectros de diferentes señales previamente moduladas pueden mezclarse y transmitirse por un mismo canal sin que se interfieran.
- **Reducción de ruido e interferencias:** Mediante determinados tipos de modulación se pueden reducir considerablemente los efectos del ruido y las interferencias. Esto se consigue aumentando el ancho de banda de la transmisión del que se tenía en banda base.

3.1.1. Modulación digital

Según [63], la modulación digital es el proceso en el cual símbolos digitales son transformados en formas de onda que son compatibles con las características del canal. Existen dos métodos principales de transmisión digital: la modulación banda base y la modulación pasa banda. En el caso de modulación en banda base sus formas de onda usualmente están formadas por pulsos. Pero en el caso de modulación pasa banda los pulsos de la señal modulan una señal senoidal, llamada señal portadora. Para transmisiones de radio frecuencia la señal modulada es convertida en campos electromagnéticos para su propagación al destino deseado.

Las modulaciones digitales cada día son más usadas debido a que presentan ventajas respecto a las modulaciones analógicas. Una de las razones son la mayor inmunidad frente al ruido y las interferencias, ya que los circuitos digitales solo operan en dos estados, por lo que la perturbación debe ser muy grande para cambiar el bit de un estado a otro; en cambio, en las señales analógicas, al tener una variedad infinita de valores, una perturbación menor puede modificar la forma de la señal y esta no podrá ser removida.

Las modulaciones digitales brindan la posibilidad de emplear técnicas de compresión de la información, esto permite reducir el volumen de la información y reducir el ancho de banda utilizado. Posibilitan el empleo de técnicas de cifrado, con lo que se evita el robo de información.

Para la selección de la modulación digital en cada aplicación, se deben contemplar las necesidades de esta, entre las que destacan: la máxima velocidad de transmisión posible, la probabilidad de error, la potencia de transmisión, el ancho de banda, entre otras.

3.1.1.1. Modulación digital pasa banda

La modulación pasa banda es el proceso en el cual la señal de información es convertida en una señal de onda senoidal con la modificación de una o más de las tres propiedades básicas de una señal portadora acorde a la representación correspondiente de los datos digitales. La forma general de la señal portadora es:

$$s(t) = A(t)\text{sen}(\theta(t)) \quad (3.1)$$

Donde $A(t)$ es la variación en el tiempo de la amplitud de la señal portadora. $\theta(t)$ es la variación en el tiempo del ángulo de fase de la señal portadora.

La variación en el tiempo del ángulo de fase está compuesta por (ω_0) que es la frecuencia en radianes de la portadora y $\phi(t)$ que es la fase, por lo que la ecuación 3.1 se puede escribir como la ecuación 3.2. Los términos (f) y (ω_0) serán usados para representar la frecuencia. Cuando sea usado (f) se refiere a la frecuencia en Hertz, cuando sea (ω_0) a la frecuencia en radianes por segundo, o frecuencia angular. Los parámetros están relacionados por la ecuación 3.3.

$$\theta(t) = \omega_0 t + \phi(t) \quad (3.2)$$

$$\omega_0 = 2\pi f \quad (3.3)$$

Por los que tenemos:

$$s(t) = A(t)\text{sen}[2\pi f t + \phi(t)] \quad (3.4)$$

La modulación digital paso banda recibirá su nombre según la variación de algunos de estos tres parámetros: amplitud (A), frecuencia (f) o fase (ϕ) de la señal portadora. Si la señal de información es digital, y la amplitud de la portadora se varía proporcionalmente a la señal de información, se produce una señal modulada digitalmente, llamada *modulación por conmutación de amplitud* (ASK). Si la frecuencia varía en forma proporcional a la señal de información se produce la *modulación por conmutación de frecuencia* (FSK), y si la fase varía de manera proporcional a la señal de información, se produce la *modulación por conmutación de fase* (PSK). Una variación al mismo tiempo de amplitud y fase en proporción con la señal de información resulta en la *modulación de amplitud en cuadratura* (QAM) [64].

El proceso de modulación se puede hacer más complejo con la intención de aumentar la capacidad de transmitir información sin aumentar el ancho de banda de la señal modulada. De los anteriores formatos de

modulación mencionados, existe la posibilidad de hacer un sistema M-ario. Donde M representa el número de posibles estados o combinaciones, que pueden obtenerse de un determinado número N de variables binarias. En la ecuación 3.5 se muestra la expresión matemática para determinar los niveles del sistema M-Ario necesarios, según las necesidades de la transmisión.

$$N = \log_2 M \quad (3.5)$$

Otra forma de definir a N es como número de bits por símbolo y M como cantidad de símbolos. Un símbolo está constituido por uno o más bits en dependencia del tipo de modulación.

3.2. Parámetros de las modulaciones digitales

Para una caracterización correcta de la calidad de un sistema de modulación digital, es necesario tener en cuenta una gran variedad de parámetros; dígase eficiencia espectral, relación señal a ruido, BER, probabilidad de error, diagrama de la constelación, por solo señalar algunos. A continuación, se describirán los más importantes.

3.2.1. Eficiencia espectral

Una de las características fundamentales de un canal de transmisión es el denominado ancho de banda (BW, del inglés *bandwith*), que se detalla como el rango de frecuencias que ocupa el espectro de una señal en la que se concentra la mayor parte de su potencia. Se puede calcular a partir de una señal temporal sometida al análisis de Fourier.

El ancho de banda mínimo necesario para transmitir portadoras M-Arias se determina a través de la relación entre la tasa de transmisión de bits (R_b , dada en bits/segundos) y la cantidad de estados o variaciones en la señal modulada, ver ecuación 3.6 [64]:

$$BW_{min} = \frac{R_b}{\log_2 M} = \frac{R_b}{N} [Hz] \quad (3.6)$$

La eficiencia espectral o densidad de información es uno de los muchos parámetros que define la calidad de una modulación digital y nos indica cuánta información es posible transmitir sobre un ancho de banda determinado. La eficiencia de ancho de banda como también se le conoce, se calcula mediante el cociente entre la tasa de transmisión y el ancho de banda mínimo necesario, el cual es determinado por el esquema de modulación empleado. Su definición matemática es [65]:

$$eficiencia_{BW} = \frac{R_b [bps]}{BW_{min} [Hz]} = N \quad (3.7)$$

Mientras mayor sea el valor de eficiencia, mejor aprovechado estará el ancho de banda y mejor será el sistema de modulación. En general, la eficiencia espectral se normaliza a un ancho de banda de 1 Hz y en consecuencia indica la cantidad de bits por segundos por Hertz que se pueden propagar a través del medio. En la tabla 3.1 se muestra los valores de ancho de banda mínimo y eficiencia del ancho de banda para las modulaciones digitales más conocidas.

Tabla 3.1: Resumen de parámetros de las modulaciones digitales [64]

Modulación	Codificación (bit)	Ancho de banda (Hz)	Eficiencia de ancho de banda (bps/Hz)
BPSK	1	R_b	1
QPSK	2	$R_b/2$	2
8-PSK	3	$R_b/3$	3
8-QAM	3	$R_b/3$	3
16-PSK	4	$R_b/4$	4
16-QAM	4	$R_b/4$	4

Del análisis anterior, se podría pensar que sería posible aumentar indefinidamente la capacidad de un canal con solamente incrementar el número de estados empleados, pero uno de los factores que realmente delimita la eficiencia espectral es el ruido. En presencia de ruido la capacidad máxima de un canal obedece al teorema de *Shannon-Hartley*; que establece la capacidad de un canal (con ancho de banda finito y una señal continua que sufre un ruido gaussiano) considerando todas las posibles técnicas de codificación de niveles múltiples y polifásicas.

Para indicar la capacidad máxima del canal (C) se emplea:

$$C = B \log_2 \left(1 + \frac{S}{N} \right) \quad (3.8)$$

Donde (B) es el ancho de banda en Hertz, (S) la potencia de la señal recibida y (N) potencia media del ruido.

Shannon estableció que es posible obtener una transmisión libre de errores si se considera transmitir a una tasa de transmisión menor o igual que la capacidad del canal ($BitRate \leq C$). En el caso inverso, la probabilidad del error en el receptor se incrementaría sin límite mientras se aumente la tasa. Es por esta razón que no se puede transmitir ninguna información útil por encima de la capacidad del canal [64].

3.2.2. Ruido y relación señal a ruido

El ruido eléctrico suele ser definido como cualquier señal eléctrica indeseable que perturba a una señal útil que se desea transmitir. Suele clasificarse como correlacionado y no correlacionado. La correlación implica la existencia intrínseca de una señal interferente que se transmite y sus efectos de distorsión armónica e intermodulación.

El ruido no correlacionado, es aquel cuya existencia es independiente de la transmisión de una señal y siempre se encuentra presente; sea por causas externas (perturbaciones atmosféricas, cósmicas, alteraciones artificiales, etc.) o por interferencias generadas por ruido interno (dentro de un dispositivo). El ruido interno suele subdividirse en tres clases [3]:

- **Ruido de disparo:** se debe a la llegada aleatoria de portadoras sobre un elemento a la salida de un aparato electrónico;
- **Ruido de tiempo de tránsito:** perturbaciones de las portadoras cuando pasan a través del dispositivo;
- **Ruido térmico:** producido por el movimiento cinético, térmico y aleatorio de los electrones, sobre cualquier conductor.

El ruido térmico también recibe el nombre de “Ruido Blanco”. Sus efectos son modelados como un canal de ruido blanco aditivo gaussiano o AWGN (*additive white gaussian noise*) y resulta ser proporcional al ancho de banda y a la temperatura. Matemáticamente, se define como [64]:

$$N = KTB \quad (3.9)$$

Donde N es la potencia del ruido térmico en watts, K la constante de proporcionalidad de Boltzmann (1.38×10^{-23} joules por grado kelvin), T la temperatura absoluta en grados kelvin y B el ancho de banda en Hertz.

En un determinado canal de transmisión; el ruido eléctrico, la interferencia de otros usuarios y los efectos de los distintos elementos de los circuitos, pueden provocar errores en la detección de la señal.

Existe una importante relación entre el nivel de potencia de la señal transmitida con respecto al nivel de potencia del ruido, conocido como relación señal a ruido (SNR, por sus siglas en inglés); es un parámetro de calidad en un sistema de comunicaciones normalmente evaluado en el lado del receptor. Entre más alta sea la SNR, mejor será la calidad del sistema.

La relación Señal a Ruido se expresa frecuentemente en dB como se muestra en la ecuación 3.10 [64]:

$$SNR = 10 \log_{10} \left(\frac{P_s}{P_N} \right) \text{ (dB)} \quad (3.10)$$

Donde P_s es potencia de la señal y P_N potencia del ruido ambos parámetros se expresan en watts.

3.2.3. Tasa de error de bit y probabilidad de error

En los sistemas de comunicación digital, el objetivo del receptor es seleccionar correctamente los símbolos del mensaje transmitidos fuera de un conjunto finito. La presencia de ruido de canal complica la tarea y causa errores de bit.

El término de la tasa o frecuencia de errores (BER, por sus siglas en inglés) es un registro empírico (histórico) del funcionamiento real de un sistema en cuanto a errores, donde un error corresponde a la recepción de un bit contrario al transmitido. El BER corresponde entonces a la proporción de bits errados respecto a los bits transmitidos en un determinado intervalo de tiempo [65].

En la ecuación 3.11 se determina la relación entre los bits recibidos erróneamente de los bits transmitidos originalmente.

$$BER = \frac{\text{número de bits erróneos recibidos}}{\text{número de bits transmitidos}} \quad (3.11)$$

Una importante medida de comparación entre los esquemas de modulaciones digitales es la probabilidad en error, ya que permite analizar el desempeño durante su transmisión en el medio y saber cuál transmitirá de forma más confiable la información. Existen dos medidas típicas de error en comunicaciones, las cuales son la probabilidad de error de bit (P_B) y la probabilidad de error de símbolo (P_E). La relación entre ellas para señales M-arias ortogonales, está dada por [63]:

$$\frac{P_B}{P_E} = \frac{2^{(k-1)}}{2^{(k)}-1} \quad (3.12)$$

Donde k es el número de bits por símbolo.

La probabilidad de error de bit es la expectativa teórica, o matemática, de que determinado sistema tenga una tasa de errores; representa una función de la relación de potencia de portadora a ruido (o en forma más específica, el promedio de la relación de energía por bit (Ec. 3.13) entre la densidad de potencia de ruido (Ec. 3.14)) y de la cantidad de condiciones posibles de codificación que se usan (M-aria). Una tasa de error de bits se mide y a continuación se compara con la probabilidad esperada de error, para evaluar el desempeño de un sistema [64].

$$E_b = CT_b \quad \left(\frac{J}{bit}\right) \quad (3.13)$$

$$N_0 = \frac{N}{B} \quad \left(\frac{W}{Hz}\right) \quad (3.14)$$

Donde E_b es la energía de un solo bit en *joules* por bit, T_b es el tiempo de bit en segundos, C la potencia de la portadora en *watts* y N_0 es la potencia de ruido normalizada a un ancho de banda de 1 Hz.

En la tabla 3.2 se ve una lista de relaciones mínimas de potencia de portadora a ruido, y de energía por bit entre densidad de potencia de ruido (E_b/N_0) que se requieren para tener una probabilidad de error de 10^{-6} , para varios esquemas de modulación.

Tabla 3.2: Comparación de presencia de errores en diversos esquemas de modulación digital ($BER = 10^{-6}$ de probabilidad de error) [64]

Técnica de modulación	Relación C/N (dB)	Relación E_b/N_0 (dB)
BPSK	10.6	10.6
QPSK	13.6	10.6
4-QAM	13.6	10.6
8-QAM	17.6	10.6
8-PSK	18.5	14.0
16-PSK	24.3	18.3
16-QAM	20.5	14.5
32-QAM	24.4	17.4
64-QAM	26.6	18.8

Como regla general para el cálculo de la probabilidad de error en sistemas digitales se asume que todos los símbolos se transmiten con la misma probabilidad (son equiprobables) y el canal de comunicaciones solo es afectado por el ruido blanco gaussiano aditivo (AWGN, por sus siglas en inglés), ya que este modelo establece la probabilidad bajo condiciones generales de que se presente algún fenómeno aleatorio, en este caso el ruido (el ruido térmico tiene una distribución gaussiana). Además, se plantea que no hay efectos de degradación en el canal, con la finalidad de establecer un caso general.

En consideración a que se modela AWGN, en las expresiones de cálculo aparece la función de error complementario (*erfc*) (Apéndice A.5). En probabilidades, una variable aleatoria x está distribuida normalmente o en forma gaussiana, si su función de densidad de probabilidad (PDF) es la curva de Gauss [66] y está dada por la ecuación 3.15:

$$P_x(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (3.15)$$

Donde μ es la media y σ es la desviación de la variable aleatoria x . En este caso σ se conoce con el nombre de desviación normal, mientras que σ^2 es la varianza o valor eficaz de la variable aleatoria x .

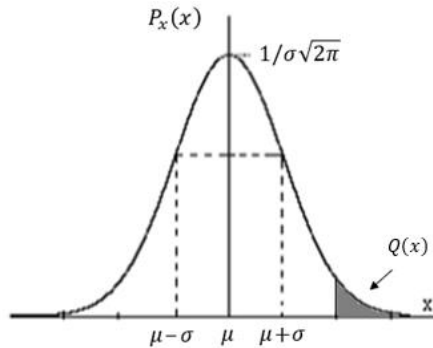


Figura 3.1: Función de densidad de probabilidad gaussiana [66].

La función de distribución acumulativa, que da la probabilidad que una variable pueda tomar un valor menor a x . Para media nula ($\mu=0$) y varianza la unidad ($\sigma^2=1$), se denota por:

$$\Phi(x) = P(X \leq x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) dt = \frac{1}{2} \left[1 + \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \quad (3.16)$$

Pero como se desea la probabilidad de que la variable tome un valor mayor a x ($P(X > x)$), la expresión final sería la siguiente:

$$P(X > x) = 1 - P(X \leq x) = Q(x) = \frac{1}{2} \operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right) \quad (3.17)$$

Donde *erf* es la función de error y *erfc* la función de error complementario. Las propiedades más importantes de la función Q (solo puede ser evaluada por tablas) son:

$$1. Q(-x) = 1 - Q(x) \quad 2. Q(0) = 0.5 \quad 3. Q(\infty) = 0$$

Para valores arbitrarios de μ y σ^2 la probabilidad de error a partir de la función Q queda descrita de la siguiente forma [66]:

$$P(X > x) = Q\left(\frac{x-\mu}{\sigma}\right) \quad (3.18)$$

En la figura 3.2 se muestran las probabilidades existentes en un sistema de dos símbolos equiprobables, donde $p(x|s_1)$ y $p(x|s_2)$ representa la probabilidad de recibir el símbolo s_1 y s_2 respectivamente y $P(e|s_1)$ y $P(e|s_2)$ son las probabilidades de error para el símbolo s_1 y s_2 .

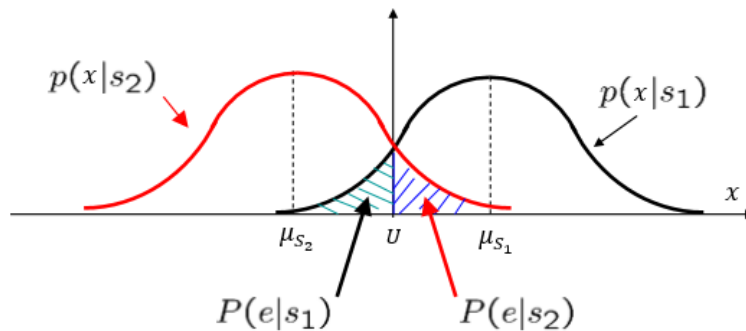


Figura 3.2: Región de decisión y probabilidad de error de símbolo.

Finalmente, la probabilidad de error de s_1 y s_2 es la siguiente [66]:

$$P(e|s_1) = P(X \leq x) = 1 - Q\left(\frac{x-\mu}{\sigma}\right) = Q\left(\frac{\mu-x}{\sigma}\right) \quad (3.19)$$

$$P(e|s_2) = P(X > x) = Q\left(\frac{x-\mu}{\sigma}\right) \quad (3.20)$$

En conclusión, la probabilidad de error (de bit o símbolo) es la estimación matemática sobre un evento que aún no ocurre, en el caso de un sistema binario de comunicaciones. La ocurrencia de un error al momento que el receptor procede a estimar si la información contenida en la forma de onda que recibe corresponde a un uno o un cero. En cambio, la tasa de errores de bit es una métrica histórica, más relativa a resultados estadísticos y se relaciona con la frecuencia de ocurrencia de un evento, en este caso, la ocurrencia de un error. La probabilidad de error se calcula con base en la estimación y el BER se define con base a la observación.

3.2.4. Tasa de error de modulación

La tasa de error de modulación (MER, *modulation error ratio*) es un factor de eficiencia de suma importancia, ya que expresa la cantidad de dispersión que sufren los máximos fasoriales reales (transmitidos o recibidos), respecto a una determinada constelación ideal. Esta es una herramienta cuantitativa que permite valorar que tan buena es una señal modulada digitalmente. El MER aporta información importante pues combina los efectos del ruido, errores de fase, error de cuadratura, y demás [65].

En el mundo real las señales siempre sufren algún tipo de deterioro, provocando que los símbolos se dispersen alrededor de su posición ideal dentro de la constelación. En la figura 3.3 se muestra de forma gráfica, un indicador de calidad de la señal modulada. El término I/Q es una abreviatura para en fase y en cuadratura, y representa un plano complejo que determina la posición de los estados de la señal en términos de amplitud y fase de acuerdo al esquema de modulación (representación los máximos fasoriales). Se muestra el vector que define un símbolo de referencia (el símbolo ideal que se quiere transmitir o recibir) y el vector del símbolo transmitido (o recibido) que es ligeramente diferente del ideal. El error de modulación es el vector diferencia entre la posición ideal del símbolo en la constelación frente al valor detectado en condiciones no ideales de interferencia, cuantificado como el módulo del vector diferencia entre ambos. [67].

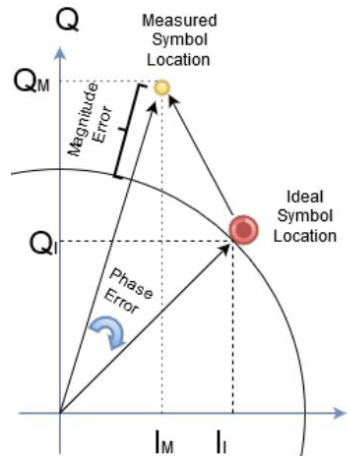


Figura 3.3: Representación gráfica del MER [3]

Matemáticamente, una definición más precisa del MER (en dB) es la siguiente [67]:

$$MER = 10 \log_{10} \left(\frac{\sum_{j=1}^N (I_j^2 + Q_j^2)}{\sum_{j=1}^N (\delta I_j^2 + \delta Q_j^2)} \right) \quad (3.21)$$

Donde I y Q son la parte real (I , *in-phase*) e imaginaria (Q , *quadrature*) de cada muestra ideal del vector del símbolo referencia, δI y δQ son la parte real e imaginaria del vector error de modulación y n es la cantidad de símbolos de la constelación. Esta definición asume que un gran número de muestras es tomado y por esto todos los símbolos de la constelación tienen la misma probabilidad de ocurrir. De dichas definiciones se deduce que el valor del MER es directamente proporcional a la calidad de una señal digital (mientras mayor sea el MER mejor).

3.2.5. Diagrama de constelación

El diagrama de constelación, también conocido como diagrama de estado-espacio de señal, es un método de representación en el plano complejo de los estados de símbolo en términos de amplitud y fase en los esquemas de modulación digital a través de diagramas polares. En ellos, se muestra un plano complejo de las muestras instantáneas de un determinado esquema; los cambios de amplitud modifican la magnitud del fasor y los cambios de fase y frecuencia modifican la posición angular del mismo [64]. Ver figura 3.4.

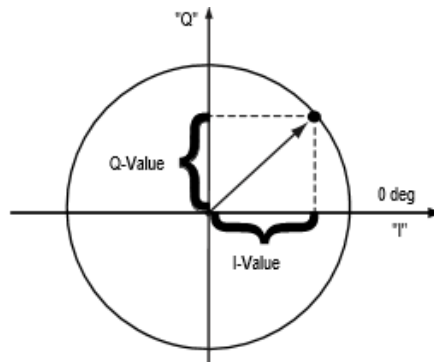


Figura 3.4: Representación de un símbolo en el diagrama de constelación.

Típicamente el eje horizontal se refiere a los componentes de los símbolos que están en fase con la señal portadora (I , *in-phase*) y el eje vertical a los componentes en cuadratura (Q , *quadrature*) (90° de desfase). En el diagrama de constelación solo se muestran las posiciones relativas de los picos de los fasores.

Este tipo de diagramas son particularmente útiles porque representan la forma de operación de los “*I/Q Modulators*”; en ellos, las señales de salida del oscilador local son separadas en dos señales desfasadas por 90° ,

para posteriormente ser combinadas y transmitidas [3]. Además, permiten visualizar el rango de error que se tiene entre los diferentes símbolos de la constelación, el cual estará definido por la regla de decisión para el detector, esta se establece en términos de regiones de decisión. Es decir, del diagrama de constelación se puede analizar qué tan tolerante será un sistema a las interferencias.

El diagrama de constelación también juega un rol muy importante en el diseño del codificador en un sistema digital de comunicaciones. La elección de la constelación determina el compromiso entre prestaciones (probabilidades de error de símbolo y de bit) y consumo de energía (energía asociada a cada símbolo y energía media por símbolo transmitido, E_s). Conviene recordar que las prestaciones dependen fundamentalmente de las distancias entre los puntos (representación vectorial de las señales) de la constelación, y en particular, de la mínima distancia entre dos símbolos. Por otro lado, la energía asociada a un símbolo (y a la señal que se utiliza para transmitirlo) está dada por la norma al cuadrado de su representación vectorial (que se corresponde a la distancia al cuadrado del vector al origen de coordenadas).

3.3. Modulaciones de referencia para el desarrollo de BPSK^{+S}

3.3.1. Modulación OOK

La modulación por desplazamiento de amplitud (ASK) es una forma de modulación en la cual se representan los datos digitales como variaciones de amplitud de la onda portadora, manteniendo constante la fase y frecuencia.

Aunque se pueden definir varios niveles de amplitud para convertir a ASK en una modulación multisímbolo, la dependencia del decisor de la potencia de entrada complica la definición de varias regiones de decisión. Por tanto, y al ser la ASK utilizada principalmente por su sencillez, no suele verse como modulación multisímbolo.

En la figura 3.5 se muestra un caso especial de la modulación ASK: la modulación de encendido-apagado (OOK, *on-off keying*), que representa los datos digitales a través de la presencia o ausencia de la señal portadora (la señal modulada es igual a la señal portadora para el caso del bit «1» y ausencia de señal para el bit «0»). La expresión general de la señal modulada es la siguiente [68]:

$$S_i = [b_i][A \cos(2\pi f_c t) + \emptyset] \quad (3.22)$$

Donde b_i es la señal binaria moduladora con dos posibles estados: «1» y «0», $A = \sqrt{\frac{2E_i(t)}{T_b}}$ la amplitud de la portadora con $E_i(t)$ la energía de bit en *joules* y T_b el tiempo de bit, f_c la frecuencia de la portadora, en general se verifica que $f_c \gg f_b = \frac{1}{T_b}$ con f_b es la frecuencia de señalización, finalmente se obtiene [68]:

$$S_i(t) = [b_i] \left[\sqrt{\frac{2E_i(t)}{T_b}} \cos(2\pi f_c t + \emptyset) \right] \quad 0 \leq t \leq T \quad i = 1, \dots, M \quad (3.23)$$

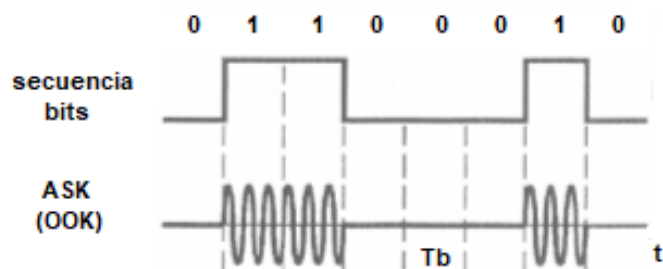


Figura 3.5: Forma de onda de salida de un modulador OOK para $M=2$.

Los símbolos de OOK basados en la ecuación 3.23 y para $M = 2$, quedan representados por (Ec. 3.24) y (Ec. 3.25):

$$S_1(t) = \sqrt{\frac{2E}{T_b}} \cos(2\pi f_c t) \quad \text{representa el «1» binario} \quad (3.24)$$

$$S_2(t) = 0 \quad \text{representa el «0» binario} \quad (3.25)$$

En la figura 3.6 se muestra el diagrama de constelación con un símbolo de amplitud cero (intersección de los ejes), y otro con la amplitud de la señal portadora. La representación se realiza en un solo eje ya que no hay variación de fase de la portadora, sino que se trata únicamente de una modificación de la amplitud. En una representación real, las pequeñas variaciones de fase o de amplitud debidas al ruido se plasmarían en un desplazamiento o ensanchamiento de los puntos que representan los símbolos.

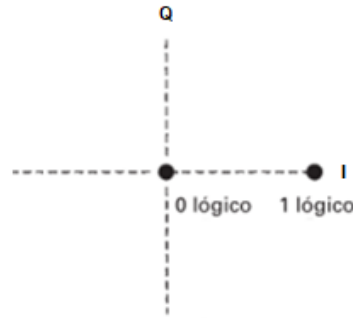


Figura 3.6: Diagrama de constelación para un modulador OOK.

De las ecuaciones 3.24 y 3.25 se aprecia que solo existe una función base de energía, dada por [69]:

$$\varphi_1(t) = \sqrt{\frac{2}{T_b}} \cos(2\pi f_c t) \quad 0 \leq t \leq T_b \quad (3.26)$$

Quedando:

$$\left. \begin{array}{l} S_1(t) = \sqrt{2E_b} \varphi_1(t) \\ S_2(t) = 0 \end{array} \right\} \quad 0 \leq t \leq T_b \quad (3.27)$$

El receptor usa la siguiente regla de decisión:

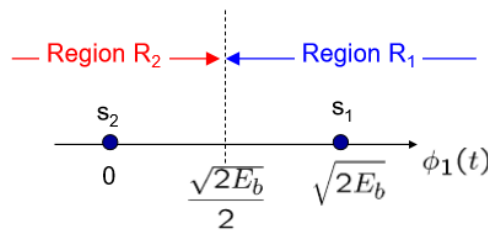


Figura 3.7: Región de decisión en modulación OOK.

La probabilidad de bit en error (P_b) para un detector OOK, está dada por [68]:

$$P_b = Q\left(\sqrt{\frac{E_b}{N_0}}\right) = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{2N_0}}\right) \quad (3.28)$$

Donde *erfc* es la función de error complementario, cuya evaluación no es trivial, pero con la ayuda de herramientas computacionales, se facilita mucho su uso.

Las ventajas de este tipo de modulación son el sencillo diseño e implementación a nivel de *hardware* (menor costo) y el bajo consumo, especialmente si se utiliza el método o modulación OOK. La desventaja es la fragilidad en presencia de interferencias por ruido eléctrico, la distancia y otros factores que afectan a la amplitud de la

señal, que pueden provocar errores en los datos recibidos. Además, presenta bajas tasas de transmisión y problemas de sincronización de señal.

El uso de esta modulación es limitado, pero se encuentran aplicaciones interesantes en las alarmas de automóviles, *tags* o membretes electrónicos usados en plazas de peaje de pago automático y cierres electrónicos por telecomandos; en estos casos se usan receptores superregenerativos de unos pocos transistores, y por ello, de muy bajo consumo y muy alta ganancia, que son capaces de demodular este tipo de señales. También se suelen utilizar en comunicaciones ópticas, ya que las fibras ópticas permiten trabajar con menores relaciones señal a ruido.

3.3.2. Modulación BPSK

La modulación por desplazamiento de fase (PSK, por sus siglas en inglés) modula la fase de una señal portadora de acuerdo con una moduladora. Las modulaciones por frecuencia y fase están relacionadas al ser sistemas de modulación angular con amplitud constante (un cambio constante de frecuencia de 1 Hz implicaría que la fase está avanzando a una velocidad de 360 grados por segundo). A pesar de ser razonablemente similar a la modulación analógica por fase convencional (PM, *phase modulation*), la señal moduladora (al ser digital), hace que la portadora cuente con un número limitado de estados de fase. A pesar de ser más resistente a las degradaciones que afectan a ASK y de no tener las limitaciones de ancho de banda de FSK, PSK se ve limitado por el número de fases que un equipo puede detectar, este factor limita su *bit rate* potencial [3].

Existen varios formatos de PSK, clasificados en dos grupos distintos; las modulaciones PSK convencionales en donde la información es definida por el estado propio de la fase y los sistemas *differential* PSK en donde son los cambios de estado, los que definen la información.

Una señal PSK es representada generalmente como [68]:

$$S_i(t) = \sqrt{\frac{2E}{T}} \cos(2\pi f_c t + \phi_i(t)) \quad 0 \leq t \leq T \quad i = 1, \dots, M \quad (3.29)$$

Donde E es la energía del símbolo en *joules*, T es la duración del símbolo en segundos, $f_c = n_c/T$ con n_c un número entero que representa el número de oscilaciones producidas en la duración de un símbolo, y $\phi_i(t)$ es la fase de la señal portadora, que puede tener M valores discretos, típicamente dados por:

$$\phi_i(t) = \frac{2\pi i}{M} \quad i = 1, \dots, M \quad (3.30)$$

Dentro de las modulaciones PSK convencionales encontramos la modulación por desplazamiento de fase binaria (BPSK). BPSK es un esquema de modulación muy sencillo pues solo presenta dos posibles fases de salida para una sola frecuencia portadora. Una fase de salida representa el «1» lógico y la otra el «0» lógico. A medida que la señal digital de entrada cambia, la fase de la portadora de salida cambia entre dos ángulos que están desfasados 180°. Se escogen estos ángulos para facilitar la demodulación, ya que es la diferencia máxima que se puede dar entre dos ángulos y así mejorar la probabilidad de error. Ver figura 3.8.

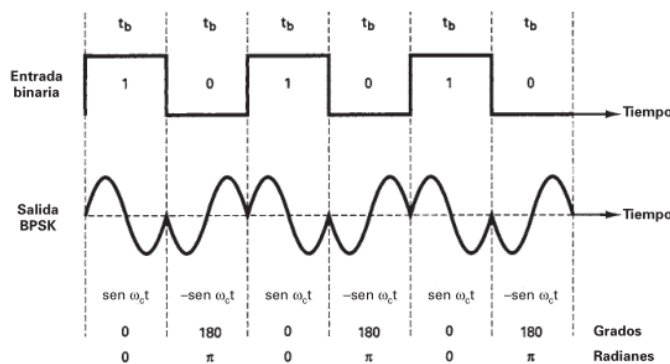


Figura 3.8: Relación de fase de salida en función del tiempo para un modulador BPSK. [64]

Para este caso solo hay dos tipos de símbolo con un bit cada uno (la energía de símbolo y el tiempo de símbolo serían iguales a la energía de bit y el tiempo de bit, $E = E_b$ y $T = T_b$), representados por (Ec. 3.31) y (Ec. 3.32) [68]:

$$S_1(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t) \quad \text{representa el «1» binario} \quad (3.31)$$

$$S_2(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \pi) = -\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t) \quad \text{representa el «0» binario} \quad (3.32)$$

El diagrama de constelación que se muestra en la figura 3.9, representa la transmisión de los dos posibles símbolos en BPSK.

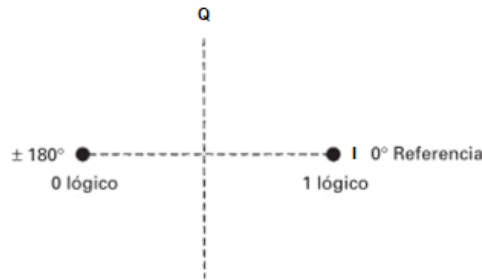


Figura 3.9: Diagrama de constelación para un modulador BPSK.

Dado que las ecuaciones 3.31 y 3.32 representan señales antipodales, solo existe una función base de energía, dada por [68]:

$$\varphi_1(t) = \sqrt{\frac{2}{T_b}} \cos(2\pi f_c t) \quad 0 \leq t \leq T_b \quad (3.33)$$

Quedando:

$$\left. \begin{aligned} S_1(t) &= \sqrt{E_b} \varphi_1(t) \\ S_2(t) &= -\sqrt{E_b} \varphi_1(t) \end{aligned} \right\} 0 \leq t \leq T_b \quad (3.34)$$

El detector del demodulador elegirá la $S_i(t)$ con la mayor correlación a la salida, llamada $z_i(t)$, o en este caso en que las señales son antipodales, usa la siguiente regla de decisión [63]:

$$S_i(t) = \begin{cases} S_1(t), & z_i(t) > 0 \\ S_2(t), & \text{otros} \end{cases}$$

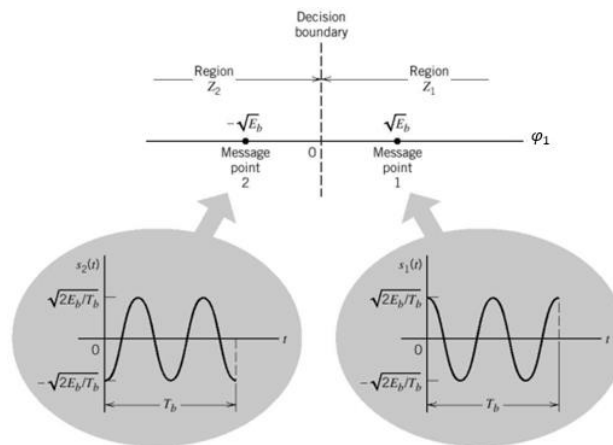


Figura 3.10: Región de decisión en modulación BPSK, para $n_c = 2$. [70]

La probabilidad de bit en error (P_b) para un detector de error mínimo binario, como es el caso de BPSK está dada por [70]:

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \quad (3.35)$$

La modulación BPSK es considerada de las más robustas debido a la gran separación de los puntos de la constelación. Dada la amplitud constante que tiene la señal portadora, esta modulación tiene la ventaja de ser menos sensible a la atenuación de la señal, además de que simplifica el diseño de los amplificadores y etapas receptoras; son precisamente estas características de simplicidad de diseño y alta eficiencia en errores las que han posibilitado su utilización en sistema de telemetría espacial, tecnologías como WI-FI y Zigbee a bajas tasas de transmisión de datos, solo por mencionar algunos ejemplos.

3.4. Técnica de codificación-modulación BPSK^{+S} (propuesta)

A diferencia de la modulación BPSK, donde solo se utilizan los estados cero y uno, BPSK^{+S} agrega un tercer estado llamado silencio (S), en un intervalo de bit (tiempo de bit) equivalente al tiempo que se necesitaría para transmitir un uno o un cero. El estado S es similar a la ausencia de portadora cuando en la modulación OOK se modulaba el «0» lógico. Para que un estado S tenga un significado en ambos extremos de la comunicación, es necesario codificar la información original (en forma binaria) a una forma triaria compuesta por los estados: «0», «1» y S.

3.4.1. Codificación BPSK^{+S}

Para explicar el funcionamiento de la codificación en BPSK^{+S}, consideremos el escenario ilustrado en la figura 3.11. Un dispositivo quiere transmitir una cadena de n bits al nodo receptor. La primera operación que BPSK^{+S} aplica sobre los datos binarios originales es su división en bloques de tamaño m bits que se codifican de manera independiente ($m = 16$, para el ejemplo). Luego se establece una ventana virtual de tamaño w bits ($w = 4$, para el ejemplo) que facilita la codificación de dos bits de la secuencia a partir de uno solo. Tanto el transmisor como el receptor deben conocer el tamaño de bloque y de ventana para un correcto proceso de codificación-decodificación. Más adelante se analiza con mayor claridad la necesidad de la ventana virtual (ventana de codificación) y el tamaño óptimo de ventana.

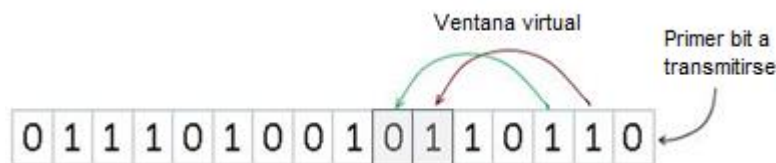


Figura 3.11: Bloque de m bits a codificar.

En el proceso de codificación se aplican una serie de reglas que permiten convertir la información binaria a triaria. Las reglas de codificación en BPSK^{+S} son las siguientes:

1. **Regla # 1:** si el codificador recibe el bit «0», a la salida aparece el estado S (sin transmisión durante un T_b).
2. **Regla # 2:** si el codificador recibe el bit «1», a la salida aparece el estado «0» o «1» dependiendo del bit contenido en w (ventana virtual) posiciones posteriores ($i + w$), este último es marcado con la bandera de bit codificado (*cod*). Si el bit en la posición $i + w$, es un cero (estado «0») y si es un uno (estado «1»). Con un solo bit transmitido el receptor recibirá la información de dos bits de la secuencia.

3. **Regla # 3:** si el codificador recibe un bit con la bandera *cod* activa, significa que este bit ya fue codificado como parte de la regla #2; se omite el bit.
4. **Regla # 4:** si el valor de la posición actual del bit (*i*) más la ventana virtual es mayor que el tamaño del bloque ($i + w > m$), se omite la aplicación de la regla #2, en caso de recibir el bit «1», a la salida aparece el estado «1».

En la figura 3.12 se muestra la aplicación de las reglas anteriores a una secuencia de bits de tamaño $m = 16$. El codificador emplea cuatro tipos de registros, necesarios para su correcta operación. El registro superior hace referencia a los bits de datos cuya posición es mostrada en el segundo registro. El tercer registro hace referencia a los bits marcados como codificados (bandera *cod*) y, finalmente, el cuarto registro muestra los estados codificados que se transmitirán. Utilizando un tamaño de ventana virtual $w = 4$, la codificación bit a bit es la siguiente:

- Bit 1 = 0: (aplicar regla #1) estado S
- Bit 2 = 1: (aplicar regla #2) bit de posición 6, estado «1»
- Bit 3 = 1: (aplicar regla #2) bit de posición 7, estado «0»
- Bit 4 = 0: (aplicar regla #1) estado S
- Bit 5 = 1: (aplicar regla #2) bit de posición 9, estado «0»
- Bit 6 = 1: (aplicar regla #3) bit ya codificado, omitir
- Bit 7 = 0: (aplicar regla #3) bit ya codificado, omitir
- Bit 8 = 1: (aplicar regla #2) bit de posición 12, estado «0»
- Bit 9 = 0: (aplicar regla #3) bit ya codificado, omitir
- Bit 10 = 0: (aplicar regla #1) estado S
- Bit 11 = 1: (aplicar regla #2) bit de posición 15, estado «1»
- Bit 12 = 0: (aplicar regla #3) bit ya codificado, omitir
- Bit 13 = 1: (aplicar regla #4) estado «1»
- Bit 14 = 1: (aplicar regla #4) estado «1»
- Bit 15 = 1: (aplicar regla #3) bit ya codificado, omitir
- Bit 16 = 0: (aplicar regla #1) estado S

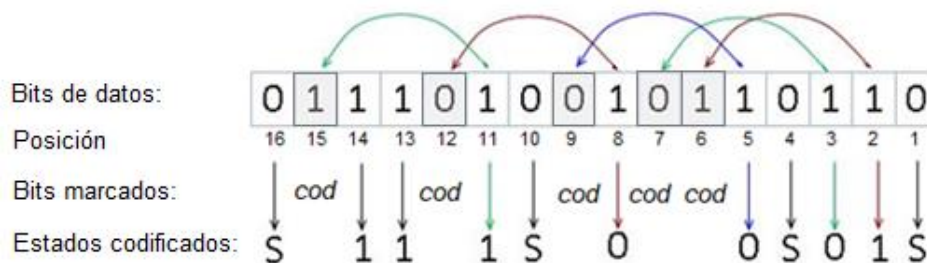


Figura 3.12: Registros del codificador BPSK^{+S}.

Del ejemplo anterior se observa que, del total de 16 bits a codificar, la técnica de codificación empleada en BPSK^{+S} los reduce a solo 11 estados («0», «1» y S), representando una disminución del 31.25 % del total de bits. De estos 11 estados resultantes, es importante destacar, que en total 4 son el estado S o sin transmisión (silencio), por lo que solo 7 de los 16 bits del bloque representarían gasto energético en la etapa de transmisión de los datos; obteniéndose un 56.25 % de reducción. En promedio se espera que, BPSK^{+S} logre una disminución del 50 % en el número de bits finales que constituyen gasto energético en la etapa de transmisión (este aspecto se analizará detenidamente a partir de experimentos en el capítulo 5).

Al igual que en el proceso de codificación, para la correcta decodificación se aplican una serie de reglas que permiten convertir la información traria recibida a binaria. Las reglas de decodificación son las siguientes:

1. **Regla # 1:** si el decodificador recibe el estado S, a la salida aparece el bit «0» en la posición final (*f*).

2. **Regla # 2:** si el decodificador recibe el estado «0» o «1», a la salida aparece el bit «1» en la posición final (f), siempre que no se encuentre ocupada (sino se aplica regla #3 antes de continuar); y en w (ventana virtual) posiciones posteriores ($f + w$) el bit «0» o «1» dependiendo del estado recibido: estado «0» (bit cero) y estado «1» (bit uno). La posición ($f + w$) es marcada con la bandera de posición ocupada (*ocup*).
3. **Regla # 3:** si el decodificador recibe cualquiera de los tres estados («0», «1» y S) y en la posición final (f) la bandera *ocup* está activa, se omite esta posición y la posición final se actualiza al siguiente valor inmediato ($f = f + 1$). Esta regla se aplica de manera recursiva hasta encontrar una posición desocupada o hasta el fin del tamaño del bloque de codificación (m).
4. **Regla # 4:** si el valor de la posición final (f) más la ventana virtual (w) es mayor que el tamaño del bloque ($f + w > m$), se omite la aplicación de la regla #2, en caso de recibir el estado «1», a la salida aparece el bit «1» en la posición f . Si es el estado S se aplica regla #1

La figura 3.13 muestra los cinco registros presentes en el decodificador: los registros posición final (f) y posición actual que representan la posición final de los bits decodificados y la posición de los datos codificados que llegan al decodificador (contenidos en el registro estados recibidos) respectivamente, el registro posiciones ocupadas que contiene la bandera *ocup* y por último el registro de bits decodificados. La decodificación bit a bit es la siguiente:

- **Estado 1 = S:** (aplicar regla #1) bit «0» en posición final $f = 1$.
- **Estado 2 = 1:** (aplicar regla #2) bit «1» en posición final $f = 2$, bit «1» en posición $f + w = 6$.
- **Estado 3 = 0:** (aplicar regla #2) bit «1» en posición final $f = 3$, bit «0» en posición $f + w = 7$.
- **Estado 4 = S:** (aplicar regla #1) bit «0» en posición final $f = 4$.
- **Estado 5 = 0:** (aplicar regla #2) bit «1» en posición final $f = 5$, bit «0» en posición $f + w = 9$.
- **Estado 6 = 0:** (aplicar regla #3 y luego regla #2) bit «1» en posición final $f = 8$, bit «0» en posición $f + w = 12$.
- **Estado 7 = S:** (aplicar regla #3 y luego regla #1) bit «0» en posición final $f = 10$.
- **Estado 8 = 1:** (aplicar regla #3 y luego regla #2) bit «1» en posición final $f = 11$, bit «1» en posición $f + w = 15$.
- **Estado 9 = 1:** (aplicar regla #3 y regla #4) bit «1» en posición final $f = 13$.
- **Estado 10 = 1:** (aplicar regla #3 y regla #4) bit «1» en posición final $f = 14$.
- **Estado 11 = S:** (aplicar regla #3 y regla #1) bit «0» en posición final $f = 16$.

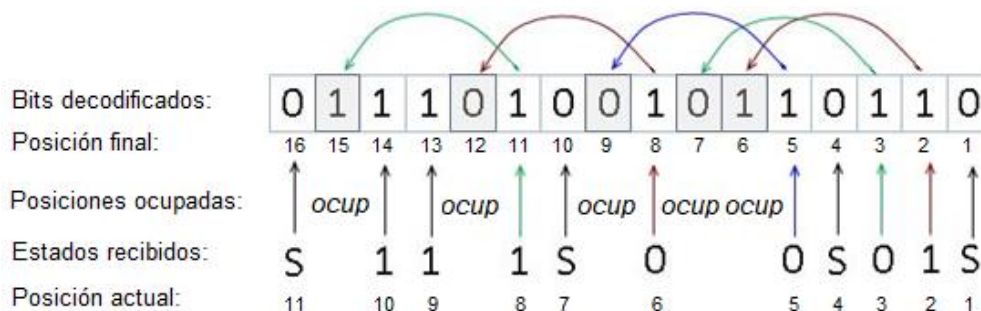


Figura 3.13: Registros del decodificador BPSK^{+S}.

3.4.1.1. Error de bit en avalancha

En una modulación tradicional como la BSPK, la relación de bit en error es uno a uno, significa que, si el ruido en el canal afecta a un bit transmitido, solo este será detectado en el receptor como un error, sin embargo, con el algoritmo de codificación empleado en BPSK^{+S}, un fenómeno denominado efecto de avalancha (BERA,

bit error rate avalanche), sucede. En la figura 3.14 se muestra el peor caso de error: cambio del estado «0» o «1» por el estado S; donde al igual que el caso contrario de cambio del estado S por los estados «0» o «1» se produce el efecto BERA.

El efecto BERA complica enormemente la aplicación de un código tradicional de corrección de errores que evite tener que solicitar la retransmisión de paquetes dañados. La manera más efectiva de reducirlo es mediante la selección de un tamaño de bloque y ventana adecuado (en el capítulo 5 se trata con detenimiento este vital aspecto). Una posible solución sería establecer el mínimo valor de tamaño de ventana de codificación en canales muy ruidosos ($w=0$).

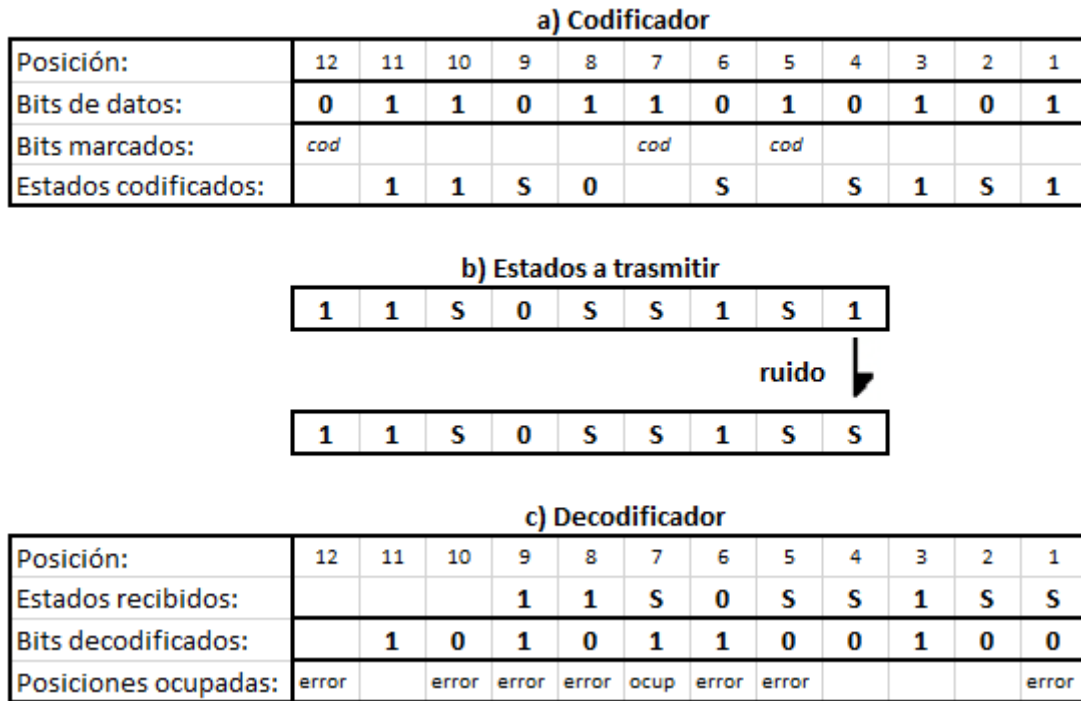


Figura 3.14: Efecto BERA.

3.4.2. Modulación BPSK^{+S}

La modulación por desplazamiento de fase binaria más silencios (BPSK^{+S}) modula la fase y la amplitud de una portadora analógica a partir de los tres estados de la señal digital triaria obtenida como resultado de la codificación BPSK^{+S}: «0», «1» y S. Los estados «0» y «1» son modulados como si se tratara de la modulación BPSK convencional; el estado S se modula a través de la ausencia de la señal portadora durante un tiempo de bit (similar a modular el cero lógico en la modulación OOK).

La expresión general de la señal modulada BPSK^{+S} para los estados «1» y «0», es la misma de BPSK (ec. 3.36) (ec. 3.38). Y para el estado S, la ecuación 3.37, representativa de OOK. Finalmente los tres estados quedan expresados de la siguiente forma:

$$S_1(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t) \quad \text{representa el estado «1»} \quad (3.36)$$

$$S_2(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \phi) = 0 \quad \text{representa el estado S} \quad (3.37)$$

$$S_3(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \pi) = -\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t) \quad \text{representa el estado «0»} \quad (3.38)$$

Las figuras 3.15 y 3.16 muestra la salida del modulador BPSK^{+S} para una secuencia triaria arbitraria y el diagrama de la constelación respectivamente.

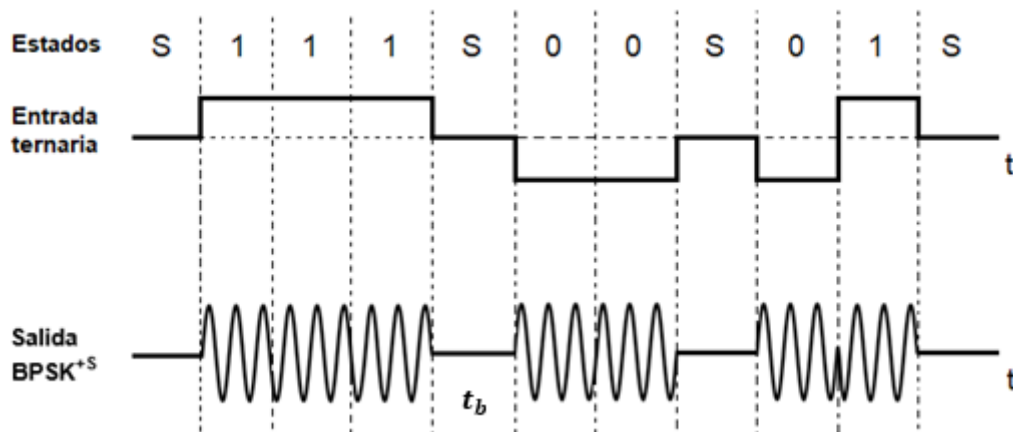


Figura 3.15: Forma de onda de salida de un modulador BPSK^{+S}.

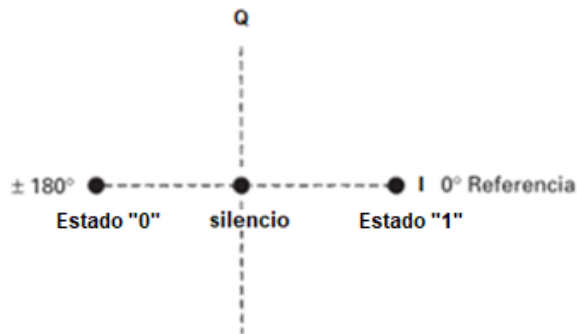


Figura 3.16: Diagrama de constelación para un modulador BPSK^{+S}.

3.4.3. Probabilidad de error de bit en BPSK^{+S}

Como se mencionó anteriormente, la modulación BPSK^{+S} tiene tres estados diferentes: cero, silencio y uno. Al obtener las funciones de densidad de probabilidad para los tres estados asumiendo ruido AWGN con varianza $\sigma^2 = N_0/2$ (es un proceso idealizado con densidad espectral de potencia constante $N_0/2$ para todas las frecuencias), se puede observar que la campana de Gauss que pertenece al estado S es más grande que las de los estados «0» y «1». En una matriz de bits, solo hay dos posibilidades: estado cero y uno, suponiendo que estos dos estados son equiprobables, tenemos que la probabilidad de transmitir un cero o un uno es la misma, lo que generará un silencio (estado S) en el 50 % de las veces ($P(s_2)$). En el caso de los bits que representan el estado uno, la codificación BPSK^{+S} puede transmitir tanto el estado cero como el uno con una probabilidad del 25 % ($P(s_1) = P(s_3)$). Ver figura 3.17.

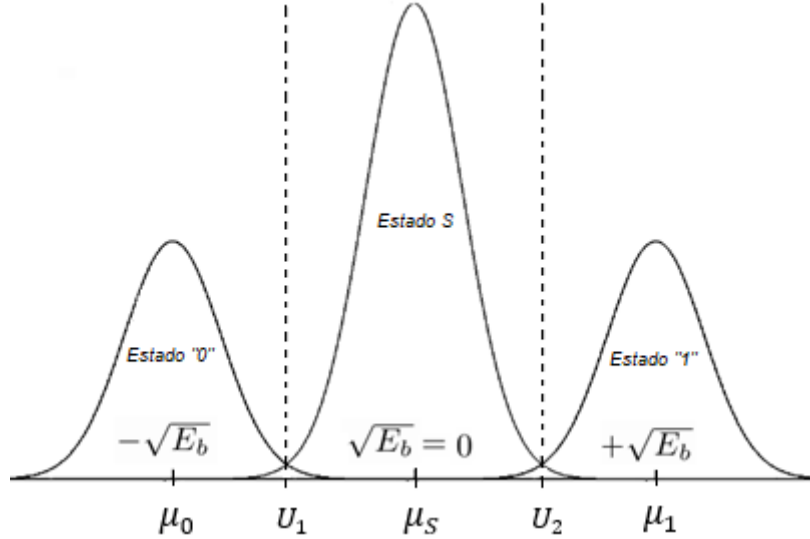


Figura 3.17: Función de densidad de probabilidad para los tres estados de BPSK^{+S}.

En la figura 3.18 se muestran los dos umbrales de decisión presentes en BPSK^{+S}. Son umbrales simétricos por lo que basta con determinar la probabilidad de error de uno de ellos para obtener la probabilidad de error total del sistema de modulación.

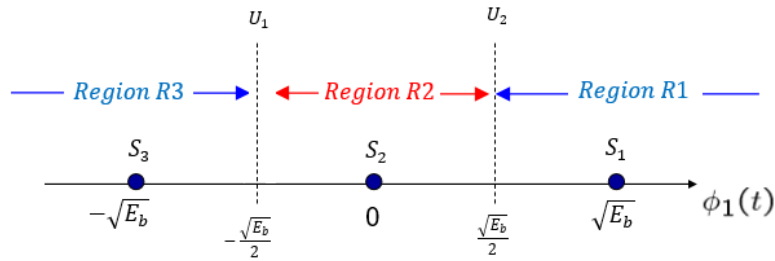


Figura 3.18: Región de decisión en modulación BPSK^{+S}.

Para el umbral de decisión U_2 las probabilidades de error de S_1 y S_2 basados en Ec. 3.19 y Ec. 3.20 son:

$$P(e|s_1) = Q\left(\frac{\mu-x}{\sigma}\right) = Q\left(\frac{\sqrt{E_b} - \frac{\sqrt{E_b}}{2}}{\frac{\sqrt{N_0}}{2}}\right) = Q\left(\frac{\frac{\sqrt{E_b}}{2}}{\frac{\sqrt{N_0}}{2}}\right) = Q\left(\frac{\sqrt{\frac{E_b}{4}}}{\sqrt{\frac{N_0}{2}}}\right) = Q\left(\sqrt{\frac{E_b}{2N_0}}\right) \quad (3.39)$$

$$P(e|s_2) = Q\left(\frac{x-\mu}{\sigma}\right) = Q\left(\frac{\frac{\sqrt{E_b}}{2} - 0}{\frac{\sqrt{N_0}}{2}}\right) = Q\left(\frac{\frac{\sqrt{E_b}}{2}}{\frac{\sqrt{N_0}}{2}}\right) = Q\left(\sqrt{\frac{E_b}{2N_0}}\right) \quad (3.40)$$

Finalmente, la probabilidad media de error (de bit y símbolo en este caso) del umbral U_2 , es:

$$P_{errorU_2} = P(s_1)P(e|s_1) + P(s_2)P(e|s_2) = 0.25Q\left(\sqrt{\frac{E_b}{2N_0}}\right) + 0.5Q\left(\sqrt{\frac{E_b}{2N_0}}\right) = \frac{3}{4}Q\left(\sqrt{\frac{E_b}{2N_0}}\right) \quad (3.41)$$

Procediendo de manera análoga para el umbral U_1 se obtiene el mismo resultado. La probabilidad de error de bit final para BPSK^{+S} es:

$$P_e = P_b = P_{errorU_1} + P_{errorU_2} = \frac{3}{4}Q\left(\sqrt{\frac{E_b}{2N_0}}\right) + \frac{3}{4}Q\left(\sqrt{\frac{E_b}{2N_0}}\right) = \frac{3}{2}Q\left(\sqrt{\frac{E_b}{2N_0}}\right) \quad (3.42)$$

Aplicando Ec. 3.17, queda:

$$P_b = \frac{3}{2}Q\left(\sqrt{\frac{E_b}{2N_0}}\right) = \frac{3}{4}erfc\left(\sqrt{\frac{E_b}{4N_0}}\right) \quad (3.43)$$

En la figura 3.19 se muestra las gráficas de probabilidad de error teórica (P_e) contra energía por bit entre densidad de potencia de ruido (E_b/N_0) para las modulaciones OOK, BPSK y BPSK^{+S}. (Apéndice A.1)

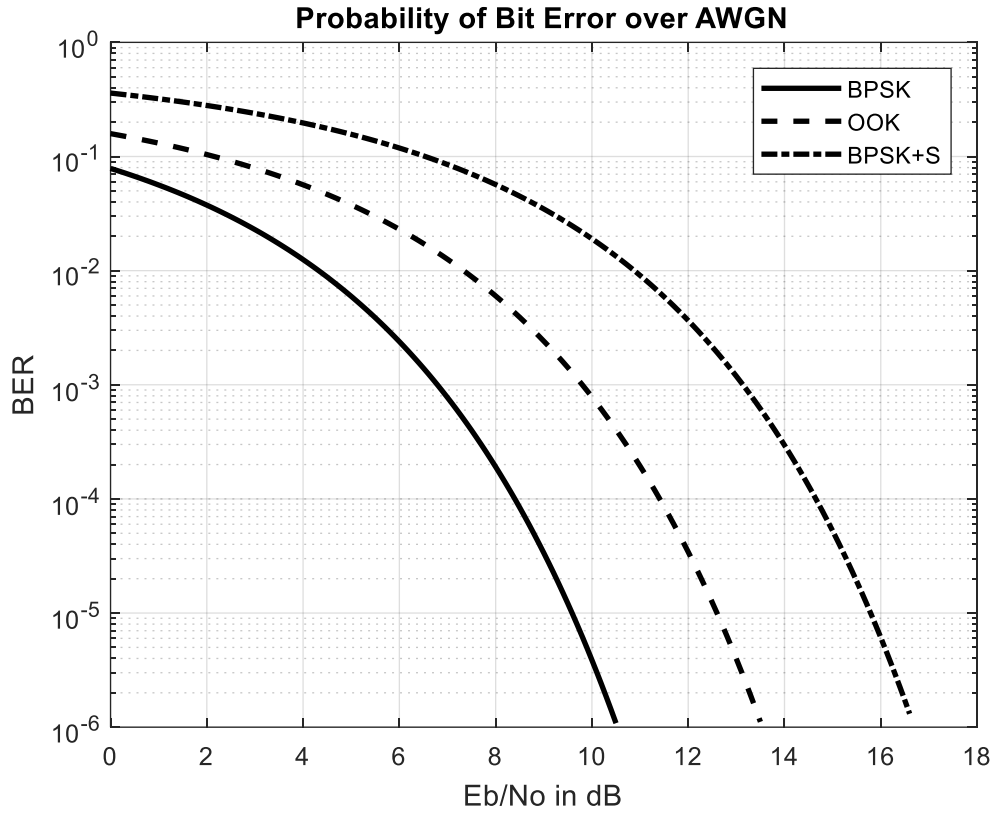


Figura 3.19: Probabilidad de error de bit en modulaciones OOK, BPSK y BPSK^{+S}.

Se observa que la probabilidad de error de BPSK^{+S} es 6.1 dB mayor que la de BPSK, lo que sin duda demuestra que la nueva técnica de codificación-modulación solo puede ser aplicada en condiciones muy específicas, pero siempre resaltando el gran ahorro energético que representa.

Implementación de BPSK^{+S} en radios NI USRP

El desarrollo teórico de la nueva técnica de codificación-modulación BPSK^{+S} necesita en la medida de lo posible una implementación que de fe de su viabilidad de uso en aplicaciones relacionadas con nodos sensores y dispositivos del internet de las cosas. Para cumplir este fin, se hace necesario el empleo de los productos desarrollados por *National Instruments*: los transceptores NI USRP, dispositivos flexibles y potentes que mediante softwares como LabVIEW son capaces de generar prototipos de comunicación por radiofrecuencia de una forma sencilla; siendo una herramienta útil para la simulación, el estudio y desarrollo de sistemas inalámbricos.

El presente capítulo se divide en dos secciones fundamentales. En la primera parte, se ofrece una introducción detallada del equipo a utilizar para la implementación, con una descripción de la arquitectura, componentes y características de los radios, así como de las herramientas de softwares utilizadas para la programación de estos. En un segundo momento se presenta la implementación realizada con la descripción detallada del módulo transmisor y el módulo receptor.

4.1. Herramientas de *hardware*

Desde su invención, los sistemas de radiocomunicación se han implementado en su totalidad en hardware, lo que hace que funcionen para un solo tipo de operación. En los sistemas de radio tradicionales, una vez que se establecen en un diseño determinados parámetros como el tipo de modulación, las bandas de transmisión y el ancho de banda, entre otros, éstos no se pueden modificar a menos que se cambien módulos de hardware.

Con los avances en la tecnología informática derivados de la Ley de Moore, existen ya herramientas programables suficientemente rápidas, compactas y baratas que hacen posible que la mayoría de los componentes de un radio se implementen a través de software, con lo que se obtiene una gran flexibilidad y por lo tanto muchas ventajas.

Por lo tanto, un radio definido por software (*software defined radio*, SDR por sus siglas en inglés) es un radio en el que todas o algunas de las funciones de la capa física (generación de señal, modulación/demodulación, codificación y sincronización) son definidas por software. Así, a través de SDR, se pueden crear transmisores o receptores para cualquier tipo de señal por medio de funciones de software o firmware que se materializan en dispositivos como arreglos de compuertas programables (*field programmable gate array*, FPGA), procesadores de señales digitales, microprocesadores y otras tecnologías programables. El uso de estas tecnologías permite agregar nuevas funciones sin necesidad de nuevo hardware.

Matt Ettus, presidente y fundador de la empresa que lleva su nombre, desarrolló una serie de productos SDR orientados a la implementación práctica de modelos inalámbricos. *Ettus ResearchTM* (adquirida por *National Instruments* en 2010), es ahora una de las compañías líderes en el desarrollo de estos equipos, incluyendo la familia de productos NI USRP.

La principal ventaja de un USRP es que mediante la conexión a una computadora toma el enfoque de un dispositivo SDR, con capacidades de procesamiento de señales digitales basados en host. Para fines de este documento se describirá y utilizará el dispositivo NI USRP 2932. Ver figura 4.1.



Figura 4.1: Radio NI USRP 2932.

4.1.1. NI USRP 2932

El NI USRP 2932 es un transceptor accesible y de fácil uso, diseñado para aplicaciones y prototipos de tipo reproducción, grabación y comunicación en la capa física, monitoreo del espectro y demás en comunicaciones inalámbricas. Este tipo de radios fueron diseñados principalmente para áreas de investigación y enseñanza, con capacidad de transmitir y recibir radiofrecuencias a lo largo de un amplio rango de frecuencias, con hasta 20MHz de ancho de banda en tiempo real. Su amplio rango de frecuencias de operación permite su uso en el estudio de aplicaciones en las bandas: ISM, radio FM, GPS, GSM y radar.

Los parámetros configurables del dispositivo para transmisión y recepción son [71]:

1. **Frecuencia de la portadora:** Configurable en un rango de frecuencias entre los 400MHz a los 4.4GHz, con una resolución de 1 kHz, para el transmisor y receptor.
2. **Ancho de banda:** Configurable hasta un máximo ancho de banda en tiempo real instantáneo de 20MHz a 16 bits ó 40MHz a 8 bits.
3. **Ganancia:** Configurable para el transmisor en un rango de 0 dB – 31 dB y con una potencia máxima de salida de 50mW a 100mW (17 dBm a 20 dBm), con resolución de 1 dB. Configurable para el receptor en un rango de 0 dB – 31.5 dB, con resolución de 0.5 dB y con una potencia máxima de entrada de 0 dBm (1mW), debido a este límite, se incluye un limitador de 0 dBm en la caja frontal de RF para evitar que ingrese una potencia excesiva en el puerto de recepción.
4. **Muestras por símbolo:** El dispositivo es configurable hasta un máximo de 25MS/s a 16 bits o 50MS/s a 8 bits (dependiendo de la configuración de la red y capacidades de la computadora del usuario, entre otros).

4.1.1.1. Arquitectura

La arquitectura de la familia NI USRP se basa en el diagrama general de cualquier radio digital, teniendo la particularidad de que sus componentes pueden ser gestionados “remotamente” vía software.

En la figura 4.2 se muestra el diagrama de bloques del modelo USRP-2932, que está separado físicamente por dos tarjetas. La tarjeta madre o *motherboard* y la tarjeta hija o *daughterboard*.

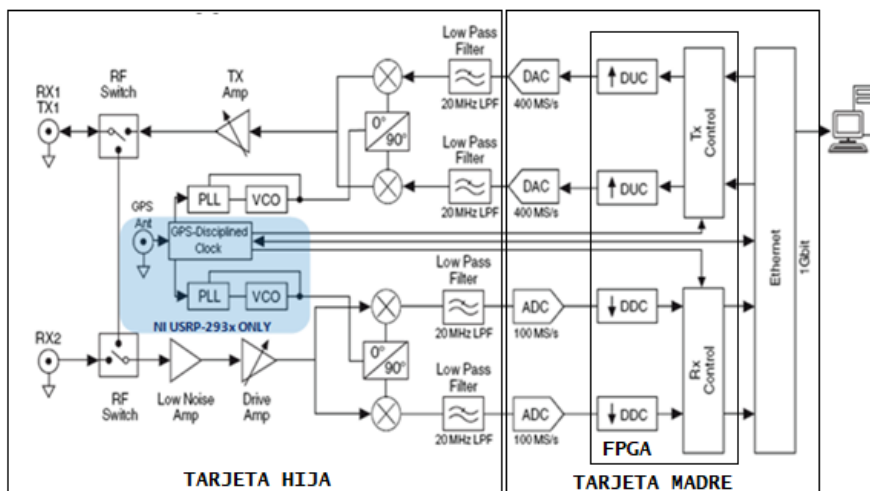


Figura 4.2: Diagrama en bloques de NI USRP 2932 [73].

La tarjeta madre es la encargada de realizar el procesamiento de la señal en banda base, posee los conversores analógico-digitales (ADC) y digitales-analógicos (DAC), realiza el proceso de decimación o diezmo de muestras y establece la conexión con un procesador externo. Su función comienza a la salida de la computadora y acaba cuando la señal atraviesa el DAC en el caso del transmisor, mientras que cuando se emplea como receptor su función empieza cuando la señal sale de la tarjeta hija y termina cuando la señal es conducida hacia la computadora, encargándose de la conversión de la señal desde el dominio digital al analógico o del analógico al digital dependiendo de cuál sea el caso y adecuando la tasa de muestreo para su transmisión [72].

En el caso de que se desee transmitir una señal, esta será generada vía software y enviada mediante la conexión Gigabit Ethernet hacia el USRP entrelazando las componentes de fase y cuadratura de la señal; del puerto Ethernet del dispositivo es conducida hacia la FPGA; esta es la encargada de desentrelazar las componentes de la señal y realizar tanto operaciones sobre el ancho de banda que se requiera mediante el uso de filtros interpoladores, desplazamientos en frecuencia de la señal haciendo uso de los *digital up converters* (DUC), finalmente la señal a la salida de la FPGA es conducida hacia el DAC de doble canal (uno para cada componente), el cual convertirá las componentes al dominio analógico para su transmisión y las enviará hacia la *daughterboard*. Para el caso del receptor, al llegar al ADC las componentes de la señal entregada por la placa hija, estas son convertidas al dominio digital y entregadas a la FPGA, quien será la encargada de desplazar estas en frecuencia mediante el *digital down converter* (DDC), de realizar operaciones sobre el ancho de banda la señal mediante filtros diezmadores y de entrelazar las componentes de la señal (I y Q) para su envío a través de la conexión hacia la computadora. El uso de los filtros interpoladores diezmadores se emplea para sincronizar las distintas tasas binarias con las que trabaja el enlace Gigabit Ethernet y el que el usuario requiere en la aplicación.

Por otro lado, la tarjeta hija es la encargada del procesamiento de la señal en radiofrecuencia y define el rango de frecuencias en las que el radio puede operar. Su función empieza a la salida del DAC y termina cuando es conducida la señal a transmitir hasta el conector SMA en el caso del transmisor, por el contrario, en el caso del receptor empieza cuando la señal llega al conector SMA y finaliza cuando ésta es conducida hacia el ADC, por lo tanto, este tipo de placa se conecta físicamente a la *motherboard*.

Existe varios tipos de tarjetas secundarias o hijas. El NI USRP 2932 utiliza el modelo SBX mostrado en la figura 4.3. La placa SBX está compuesta por un transceptor de banda ancha con osciladores independientes para el transmisor y el receptor permitiendo así operar en modo *full-duplex* con señales complejas. Por defecto esta placa opera convirtiendo la señal de radiofrecuencia directamente a banda base. Presenta una potencia típica de salida de 100 mW en transmisión y un ancho de banda de 40 MHz tanto para el transmisor como para el receptor, dicha placa está diseñada para soportar MIMO. El rango de operación es de 400 MHz-4400 MHz presentando ganancias variables tanto en la cadena del transmisor como en la del receptor. Su figura de ruido va desde 5 dB hasta 10 dB [72].



Figura 4.3: Tarjeta hija SBX [73].

Otros componentes importantes de la placa SBX son [72]:

- **Mezclador, VCO y PLL:** Representan una parte sumamente importante en la arquitectura de los radios; el mezclador es el encargado de pasar de la frecuencia intermedia a la frecuencia de radiofrecuencia, haciendo uso del VCO (*voltage controlled oscillator*) y del PLL (*phase locked loop*).
- **Amplificadores:** Existen tres tipos de amplificadores distintos. El *transmit amplifier* donde la señal recibe la ganancia necesaria para que sea transmitida (rango de 0 a 31.5 dB), el *low noise amplifier* (LNA) es una etapa de preamplificación enfocada a amplificar una señal degradada sin decrementar la relación señal a ruido y el *drive amplifier* que incrementa la señal recibida.
- **Filtros pasa-baja:** Permiten el paso de cierto intervalo de frecuencias para recuperar la frecuencia intermedia de la señal en la que trabajan los dispositivos del NI USRP.
- **GPS disciplined clock:** El reloj disciplinado GPS (*global positioning system*) permite al USRP funcionar como dispositivos sincronizados en tiempo, de fase coherente incluso cuando son separados por grandes distancias.
- **Conectores SMA:** Uno de ellos se puede configurar bien como transmisor o bien como receptor (TX/RX), mientras que el otro puerto solo puede actuar como receptor (RX2).

4.1.1.2. Antenas

Para los alcances de este estudio, se hizo uso de dos antenas monopolo distintas. La antena VERT400 es un monopolo omnidireccional tribanda cuyas bandas principales de operación se encuentran en los 144MHz (con 0dBi a $\frac{1}{4}$ de onda), los 400MHz (con 0dBi a $\frac{1}{4}$ de onda) y los 1200MHz (con 3.4dBi a $\frac{5}{8}$ de onda). Tiene una longitud física de 17 centímetros, posee un conector SMA y soporta una potencia máxima de 10 watts. Por otro lado, la antena VERT2450 es un monopolo dibanda que comparte muchas de las características de la VERT400, pero su rango de frecuencias opera de 2.4 a 2.5 GHz y de 4.9 a los 5.9 GHz [73]. Ver figura 4.4.



Figura 4.4: Antenas usadas por el NI USRP 2932 [73]

4.2. Herramientas de *software*

La tecnología SDR se caracteriza fundamentalmente por su componente *software*. El *driver* necesario para trabajar con el NI USRP puede ser descargado desde la web oficial de *National Instruments* [74]. Para una programación más amigable de los NI USRP 2932, se pueden emplear plataformas como:

- **GNU radio:** Herramienta *software*, libre y de código abierto de desarrollo que provee la posibilidad de implementar sistemas basados en *software radio* mediante el uso de bloques de procesamiento de señal. Es una herramienta de simulación que puede ser utilizado junto a hardware RF (NI USRP) para implementar físicamente sistemas de *software radio*.
- **LabVIEW:** Plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico pensado para sistemas hardware y *software* de pruebas, control y diseño, simulado o real y embebido.
- **MATLAB (*simulink*):** Entorno de programación visual de diagramas de bloques integrado en Matlab. Utilizado para la simulación de sistemas, puede ser utilizado junto a hardware USRP para implementar físicamente sistemas de *software radio*.

Mediante el *software* LabVIEW se puede interactuar con el hardware NI USRP para el desarrollo de algoritmos de comunicación que procesan las señales recibidas, así como sintetizar las señales para la transmisión. El controlador NI USRP *hardware driver*, proporciona funciones de LabVIEW para la configuración *hardware/software* con las herramientas para apertura/cierre de sesiones y operaciones de lectura/escritura.

Para el caso particular de LabVIEW, el *driver* cuenta ya con un paquete: *LabVIEW Modulation Toolkit* que proporciona un rápido desarrollo de los sistemas de comunicaciones para la simulación o para operar en las señales asociadas al hardware NI USRP. En lo que se refiere a la transmisión, se incluyen proyectos, que proporcionan funcionalidad para la generación de secuencias, la codificación de canal y modulación de banda base. Las funcionalidades para el receptor incluyen: decodificación del canal, demodulación, entre otras. Además de funcionalidades que permiten la medida del BER y medidas en el dominio de la modulación y la visualización orientada a las comunicaciones.

En LabVIEW también encontramos el módulo *MATLAB script* permite utilizar la sintaxis de los archivos *MATLAB* (.m) en un diagrama LabVIEW para combinar modelos de programación gráficos y de texto.

Para fines de nuestra implementación se utilizará la herramienta LabVIEW 2017 como *software* de procesamiento de señal con la ayuda de la programación en *MATLAB* 2017.

4.2.1. LabVIEW

El *Laboratory Virtual Instrument Engineering Workbench* o LabVIEW es un lenguaje de programación gráfico (lenguaje G) que desarrolló *National Instruments*. Permite crear código con el uso de diagramas de bloques, lo que le hace muy adecuado para el diseño y la implementación de algoritmos de comunicaciones.

Originalmente estaba orientado a aplicaciones de control de equipos electrónicos utilizados en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. Los programas creados en LabVIEW se guardan en ficheros llamados VI (*virtual instrument*) ya que su apariencia y funcionamiento imitan a los de un instrumento real. Todos los VIs tienen dos partes, una interactiva por el usuario: Panel Frontal y otra de código fuente: Diagrama de Bloques; y aceptan parámetros de otros VIs. Además de contar con paletas que contienen los objetos necesarios para implementar y desarrollar tareas [75].

El Panel Frontal muestra las salidas que vienen determinadas por las entradas que introduce el usuario al ejecutar la aplicación. Está formado por pulsadores, gráficas, botones, etc. Cada uno puede estar definido o bien como un control o como un indicador. Los primeros son objetos que se utilizan para introducir datos al programa y el usuario es el responsable de su manipulación para darle los valores deseados, son por tanto, las variables de entrada. Los indicadores por su parte se utilizan para presentar los resultados, son variables de salida por lo que el usuario no puede modificarlos.

El Diagrama de Bloques es donde se realiza la programación, es decir, contiene el código del programa para realizar el procesamiento de las entradas y salidas creadas en el Panel Frontal. En el Diagrama de Bloques se incluyen funciones y estructuras que, en el lenguaje G son nodos elementales. Son totalmente análogos a los operadores o librerías de funciones de los lenguajes convencionales. Las funciones y SubVIs realizan tareas concretas.

Por otra parte, las estructuras y cables determinan el flujo de los datos en el programa. El Panel Frontal y el Diagrama de Bloques se conectan a través de los terminales. Es decir, los controles e indicadores que se colocan en el Panel Frontal, en el Diagrama de Bloques se identifican mediante los terminales. Ver figura 4.5.

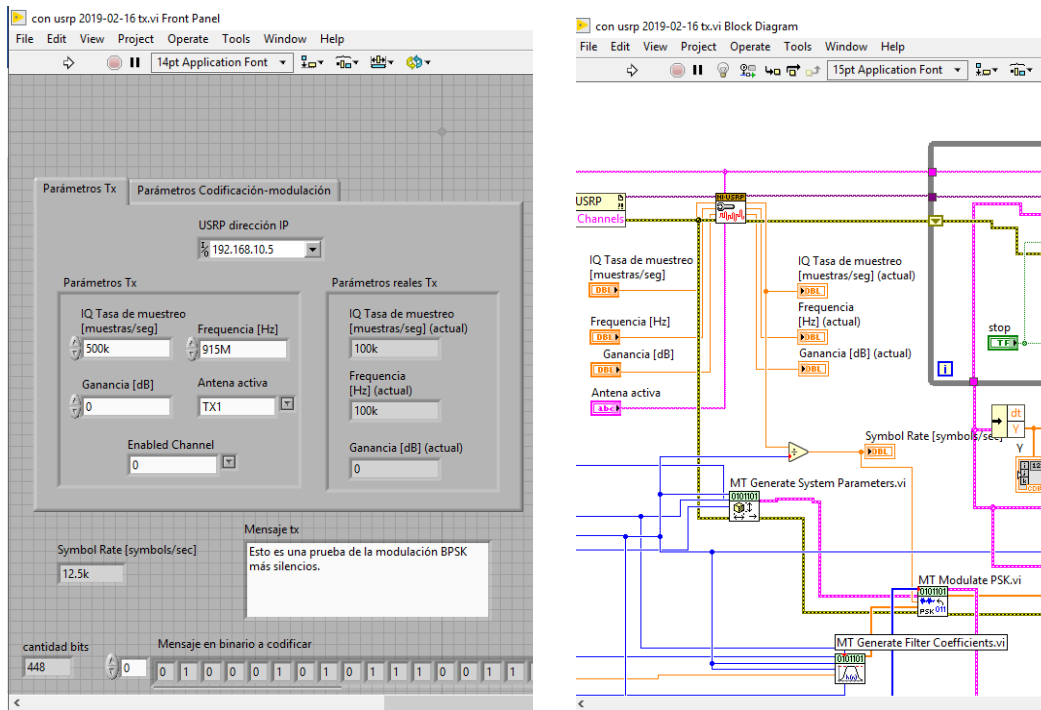


Figura 4.5: Panel frontal y diagrama de bloques.

Además, es importante destacar que existen submenús que corresponden a *toolkits* que únicamente aparecen una vez han sido instalados. Estos *toolkits* son herramientas adicionales de software que se han desarrollado con unos fines concretos y especializados que se suministran como productos por separado. Un ejemplo es el ya mencionado *LabVIEW Modulation Toolkit*.

Para auxiliarnos a la hora de programar, existe la ventana *Context Help* donde se muestra la información básica de los objetos de LabVIEW cuando se sitúa el cursor encima de cada objeto. De esta forma, se muestran los iconos de las funciones, controles, subVIs e indicadores con los cables conectados a cada terminal. Cuando se mueve el cursor encima de la caja de opciones el *Context Help* muestra una descripción de estas opciones.

En resumen, el diagrama a bloques consta de cuatro objetos importantes [71]:

- **Terminales:** ciertos bloques o íconos que pueden ser de información fija para el código, o que estarán relacionados directamente con la interfaz del usuario como son las de control e indicación.
- **Nodos:** ciertos bloques o íconos que realizan las operaciones de forma análoga a las sentencias y funciones en un programa. Estos bloques poseen entradas y/o salidas para el flujo de la información, se encuentran en la paleta de funciones y se pueden clasificar según su complejidad.
- **Estructuras:** Estos elementos controlan las veces y/o formas de ejecución del VI, como son los bucles *for*, *while*, casos estructurados, estructura por secuencia, por tiempo o por evento. Estos nodos se localizan en la sección *structures* de la paleta de funciones. Ver figura 4.6.

- Cables:** Para transferir la información entre los boques del VI. Contienen un tipo de información o varios en el caso de *clusters*. Pueden ser conectados a varias funciones o VI que puedan leer información, similar a cargar parámetros guardados en líneas de texto en lenguajes de programación. Deben estar cableadas todas las entradas requeridas de las funciones o los Vis, o el programa no funcionará adecuadamente. Los cables tienen diferentes colores, estilos y grosores dependiendo el tipo de información que lleven, similar al color de las terminales de control e indicación.

Otro aspecto no menos importante de LabVIEW son los *scripts*, estructuras que también aceptan texto, de manera que se puede combinar la programación propia de LabVIEW con la más tradicional programación textual. El *MATLAB script node* se encuentra en el menú *Mathematics > Scripts & Formulas > Script Nodes*; el nodo llama a un programa externo, específicamente el servicio *Matlab Server* a través de *ActiveX* para ejecutar comandos (sólo en el sistema operativo *Windows*). Con este tipo de *script* se facilita mucho la programación en LabVIEW y será muy utilizado en este trabajo de tesis. Ver figura 4.6.

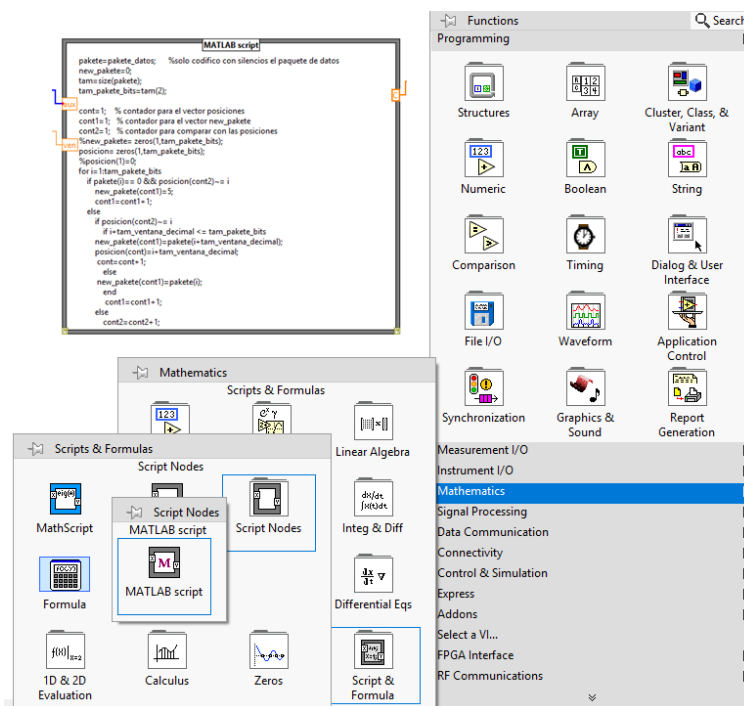


Figura 4.6: MATLAB *script node* y paleta de funciones.

La gran ventaja de LabVIEW es que puede ejecutar procesos paralelos; múltiples nodos se pueden ejecutar simultáneamente dando una ventaja apreciable en tiempo de ejecución.

4.2.1.1. LabVIEW y los radios NI USRP

El entorno de desarrollo primario para NI USRP es LabVIEW, debido a que ambas partes tanto la de hardware como la de software son desarrollados por el mismo fabricante: *National Instruments*; son innumerable las ventajas que proporciona el uso de LabVIEW como software de aplicación para interactuar con el NI-USRP 2932.

Para acceder a muchas de las características que facilitan la configuración correcta de los radios con LabVIEW es necesario instalar el NI USRP *hardware driver* como se ha explicado en secciones anteriores. Para utilizar sus funciones instaladas; desde el diagrama de bloque, en la paleta de funciones se sigue la siguiente ruta: *Instruments I/O > Instruments drivers > NI-USRP*. Ver figura 4.7.

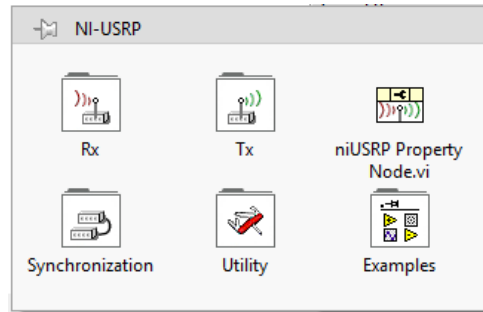


Figura 4.7: Paleta NI-USRP en LabVIEW.

Las principales funciones que utiliza LabVIEW para interactuar con el NI-USRP 2932 se agrupan en tres categorías: [76], ver tabla 4.1.

Tabla 4.1: Principales funciones de LabVIEW para comunicación con NI-USRP 2932

Modo	Categorías		
	Configuración	Lectura/escritura	Cerrado
Transmisión	niUSRP Open Tx Session.vi	niUSRP Write Tx Data (poly).vi	niUSRP Close Session.vi
	niUSRP Configure Signal.vi		
Recepción	niUSRP Open Rx Session.vi	niUSRP Fetch Rx Data (poly).vi	niUSRP Abort.vi
	niUSRP Configure Signal.vi		niUSRP Close Session.vi
	niUSRP Initiate.vi		

A continuación, se describirán las características de cada uno de estos bloques ordenados atendiendo la función que realizan [71] [77]:

- **Inicio de una sesión** (*niUSRP Open Rx Session.vi*/*niUSRP Open Tx Session.vi*): Es el VI empleado para crear una sesión entre el *software* y el dispositivo NI-USRP, reconocimiento inicial del radio. Establecen un identificador específico para cada flujo de datos que se utiliza para identificar esta sesión del instrumento en los posteriores NI-USRP VIs. Ver figura 4.8.

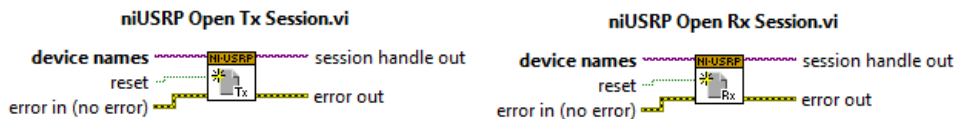


Figura 4.8: Bloques de inicio de sesión en el trasmisor y el receptor.

- **Configuración de señal** (*niUSRP Configure Signal.vi*): Este VI puntualiza los parámetros reconfigurables del dispositivo, ya sea como transmisor o receptor. Entre esos parámetros se encuentran la ganancia de la antena, la frecuencia de la portadora, el ancho de banda y el puerto de antena activo (caso de la recepción). Hay que tener en cuenta que no todos los valores en frecuencia y ganancia son válidos, estos dependen directamente de la capacidad de la tarjeta *daughterboard*. La salida del módulo ofrecerá los parámetros reales sobre los que el NI-USRP operará. Ver figura 4.9.

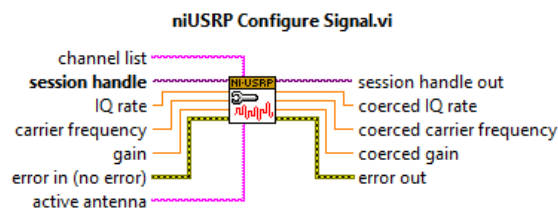


Figura 4.9: Bloque de configuración de señal.

- **Inicio de recepción de señal de radio frecuencia** (*niUSRP Initiate.vi*): Este VI inicia la sesión de recepción y le informa al NI-USRP que todas las configuraciones están completas para que comience la adquisición de IQ data. Ver figura 4.10.



Figura 4.10: Bloque inicio de recepción de señal de RF.

- **Entrega de datos** (*niUSRP Fetch Rx Data (poly).vi* / *niUSRP Write Tx Data (poly).vi*): Son módulos encargados de enviar las muestras obtenidas por el NI-USRP hacia el software de operación (LabVIEW) y viceversa. En la recepción, los datos de salida pueden ser de distinto tipo (números enteros, *doubles*, complejos y formas de onda). En la transmisión únicamente fungirá como una “puerta de datos” hacia la antena. Ver figura 4.11.

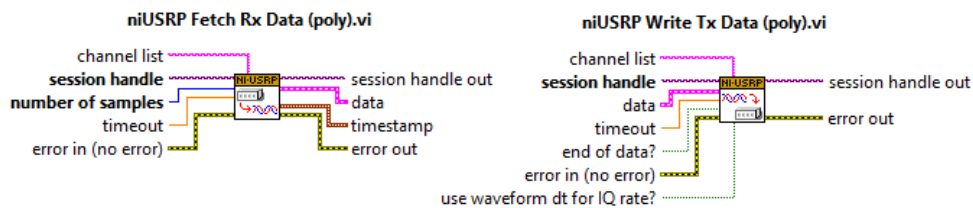


Figura 4.11: Bloques de intercambio de datos.

- **Finalizar adquisición y cerrar sesión** (*niUSRP Abort.vi* / *niUSRP Close Session.vi*): El primer módulo le dice al NI-USRP que detenga una adquisición en progreso. Permite cambiar la configuración sin cerrar por completo la sesión. Mientras el segundo módulo cierra la sesión de transmisión o recepción actual, liberando la memoria en uso por esa sesión. Después de llamar a este VI, ya no puede transmitir ni recibir datos del NI-USRP hasta que una nueva sesión sea establecida. Ver figura 4.12.

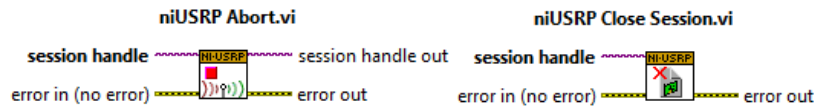


Figura 4.12: Bloques de abordar y cerrar sesión.

4.2.2. MATLAB

MATLAB (*Matrix Laboratory*) es un lenguaje de alto nivel, diseñado para proveer facilidades de cálculos numéricos, visualización y programación en un ambiente muy sencillo de utilizar. Ofrece un entorno de desarrollo integrado (IDE por sus siglas en inglés) muy versátil, con un lenguaje de programación propio (lenguaje M). Proporciona una interfase con líneas de comando para resolver problemas lineales y no lineales, además de otros experimentos en forma numérica. Utiliza una notación matemática simple para tratar los problemas y resolverlos. Su poder radica en el manejo de matrices en forma eficiente, también incluye su propio compilador lo cual permite extender su uso, permitiendo al usuario crear sus propios comandos, clases y funciones. Es compatible con uno de los más usados lenguajes de programación: C y C++ [78].

El *software* fue creado por el matemático y programador de computadoras Cleve Moler en 1984; con el propósito netamente matemático a base de álgebra lineal y análisis numérico, sin necesidad de escribir programas en dicho lenguaje. Con los años la herramienta ha sido modificada pasando por varias versiones. En cada versión se han incorporado nuevas funciones para una gran variedad de aplicaciones ingenieriles como: simulación de sistemas dinámicos, visión artificial, análisis estadístico, análisis y diseño de controles automáticos, entre otras. También es ampliamente utilizado en [78]:

- Cálculos numéricos
- Desarrollo de algoritmos
- Modelado, simulación y prueba de prototipos
- Análisis de datos, exploración y visualización
- Graficación de datos con fines científicos o de ingeniería

El entorno de trabajo de MATLAB consiste en un mosaico de ventanas de funcionalidad diversa. Múltiples ventanas pueden mantenerse abiertas y tenerse a la vista en el mosaico o superpuestas en las sub-ventanas o independientes del escritorio. Ver figura 4.13.

Las cinco ventanas principales son [78]:

1. **Command Window:** es la ventana en la que se escriben las instrucciones que se quieren ejecutar.
2. **Current Directory / Workspace:** la primera muestra el contenido de la carpeta de trabajo. La dirección de la carpeta de trabajo se puede cambiar mediante la barra desplegable que aparece encima de las ventanas. La ventana *workspace* muestra información sobre las variables y objetos definidos.
3. **Command History:** esta ventana muestra los últimos comandos (instrucciones) ejecutados.
4. **Editor / debugger:** editor y depurador de ficheros (.m). Para editar ficheros de ordenes de Matlab, ejecutarlos y seguir paso a paso su ejecución facilitando la depuración de errores.
5. **Plot Windows:** contiene ventanas gráficas (pueden abrirse varias) para mostrar las gráficas y dibujos.

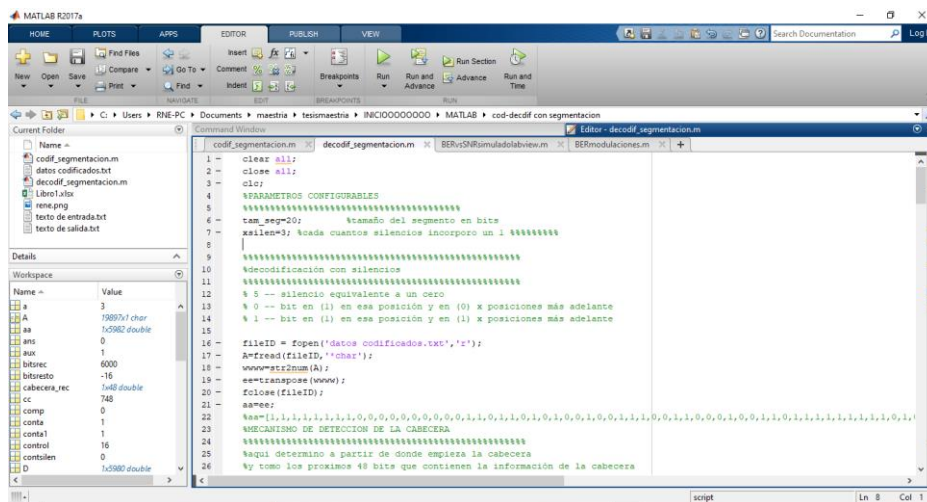


Figura 4.13: Interfaz de MATLAB.

4.2.2.1. Entorno de programación

Desde las primeras versiones MATLAB incorpora una característica muy interesante: la capacidad para programar. En efecto, es posible crear archivos que contengan las operaciones que se desean realizar. Además, permite incorporar nuevas funciones de MATLAB realizadas por el propio usuario.

Una de las características del entorno MATLAB es que permite que las instrucciones puedan ser tomadas de un archivo en lugar de ser introducidas en la ventana de comando.

La idea es simple: si el usuario va a repetir a menudo un conjunto de instrucciones puede escribirlas en un archivo de texto. Posteriormente se le indica a MATLAB que lea dicho archivo, ejecutando las instrucciones una por una. El efecto es el mismo que si el usuario las hubiese escrito en el *command window*. Esto supone un gran ahorro de tiempo. MATLAB incorpora su propio editor de archivos, llamado *M file editor*. La extensión utilizada para distinguir estos archivos generados por el editor es: (.m).

En esta tesis, se hará uso de la programación en MATLAB para el desarrollo de una herramienta que permita establecer comparaciones en el ámbito de la codificación utilizada en la nueva técnica de codificación-

modulación BPSK^{+S}, permitiendo determinar el tamaño óptimo de ventana de codificación, el grado de reducción promedio de bits a modular, el porcentaje de silencios totales en la trama codificada y el número de bits de relleno incorporados por razones de sincronización del sistema.

4.3. Implementación de BPSK^{+S}

Para la simulación del sistema de comunicación inalámbrica, basado en BPSK^{+S}, como se ha explicado en secciones anteriores, se utiliza el NI USRP 2932 con auxilio del software LabVIEW. Se diseña un transmisor y un receptor. El transmisor funcionará en base a un codificador y modulador BPSK^{+S} y el receptor mediante un demodulador y decodificador BPSK^{+S}, los cuales están constituidos por diferentes etapas; cada una de estas se irá detallando, permitiendo así obtener una correcta implementación y su entendimiento.

4.3.1. Transmisor BPSK^{+S}

El transmisor implementado tiene cinco bloques principales: configuración del NI USRP 2932, conversor del mensaje a bits, codificador BPSK^{+S}, modulador BPSK^{+S} y presentación de datos. Ver figura 4.15.

En la figura 4.14 se muestra el panel frontal del transmisor. La parte superior izquierda representa los parámetros de configuración del NI USRP y los de codificación-modulación BPSK^{+S}. La parte superior derecha representa el diagrama de la constelación. En un espacio central se muestra el mensaje a transmitir (“*Esto es una prueba de la modulación BPSK más silencios*”) y la posibilidad de escoger entre codificar o no el mensaje. En la parte inferior se muestran el mensaje binario a codificar (salida del bloque conversor del mensaje a bits) con el número total de bits y bytes; también aparece el mensaje codificado y el mensaje final a transmitir (ya incluye los bits de la cabecera) junto con las estadísticas de número de bits reducidos producto de la codificación, cantidad de silencios de la secuencia final y porcentajes de reducción en ambos casos.

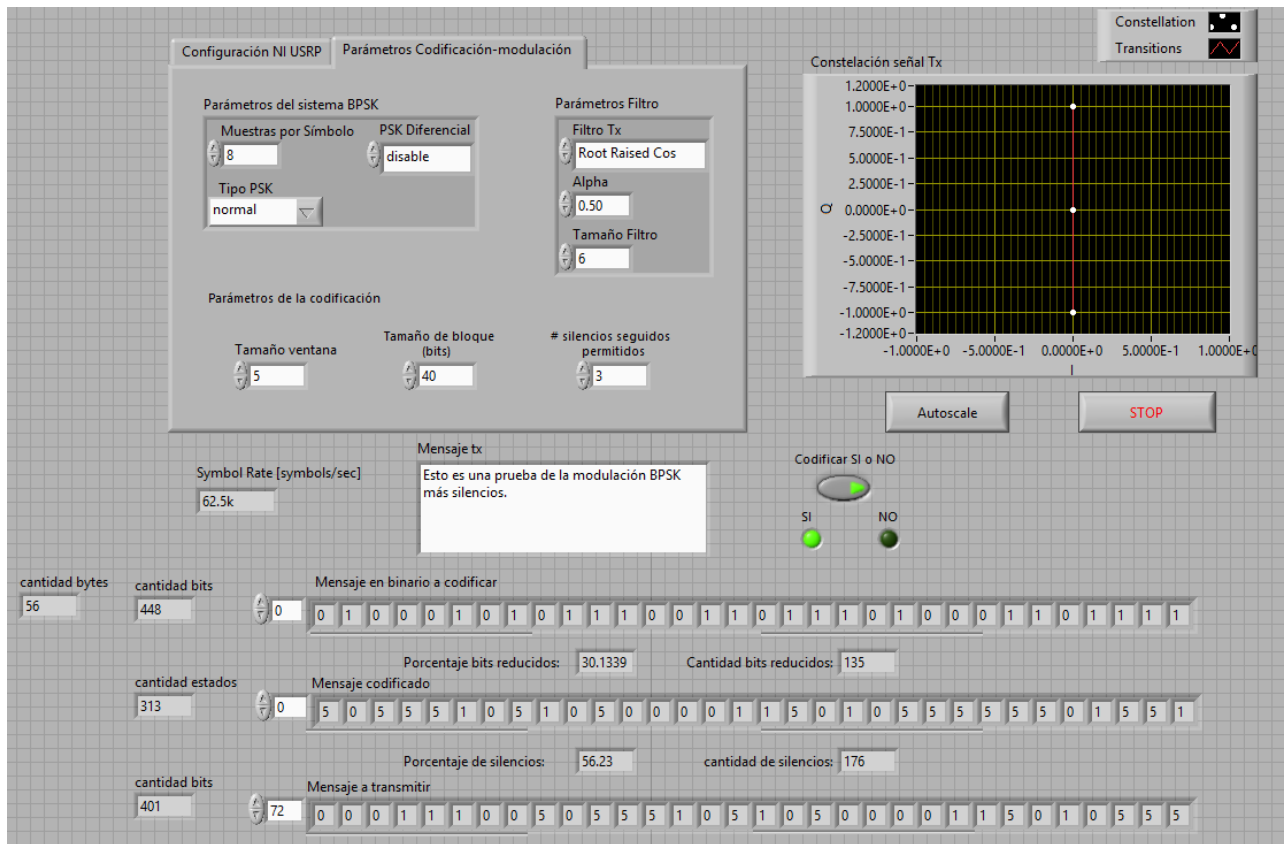


Figura 4.14: Panel Frontal transmisor BPSK^{+S} en LabVIEW.

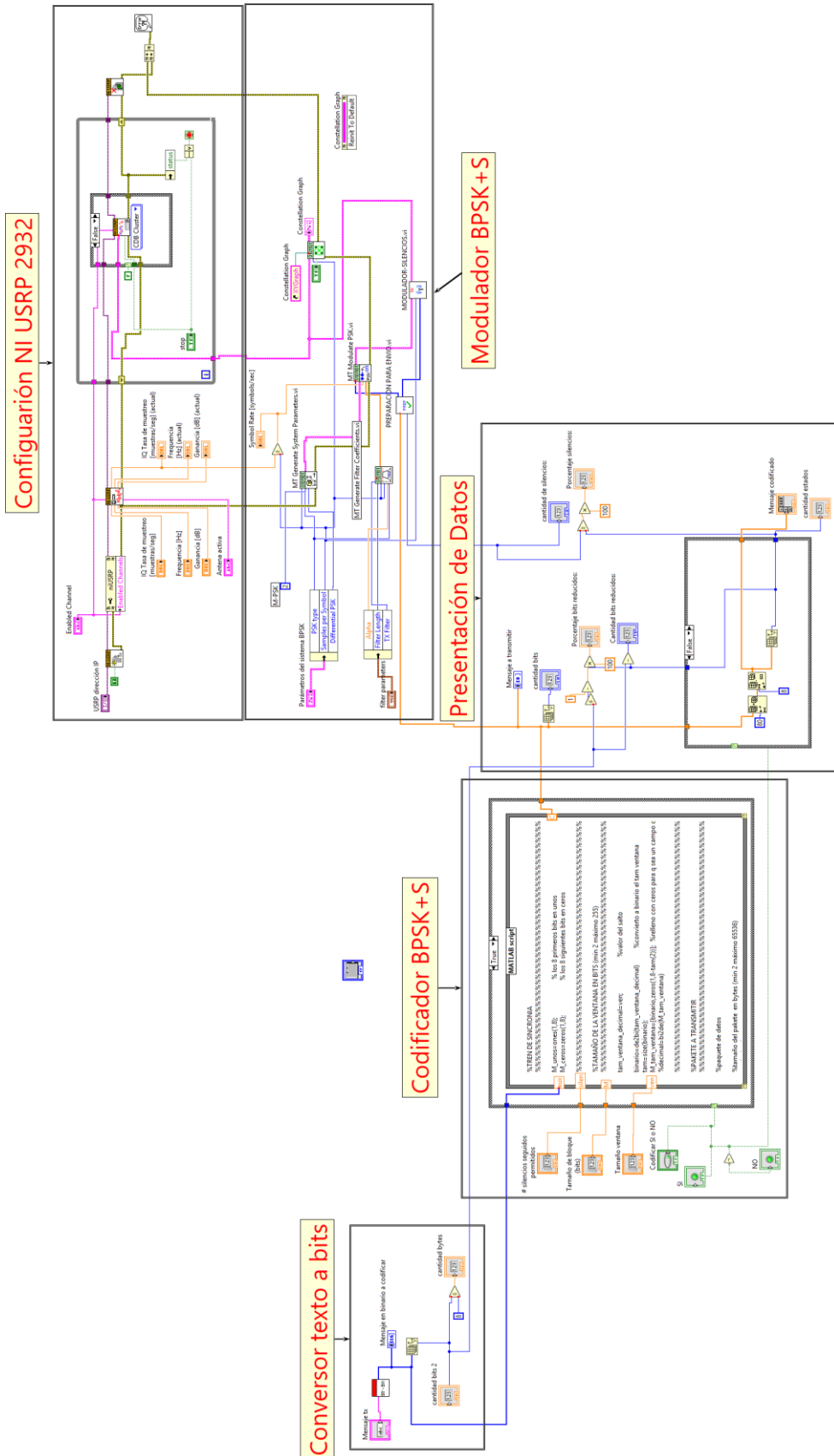


Figura 4.15: Diagrama en bloques transmisor BPSK+S en LabVIEW.

4.3.1.1. Bloque configuración NI USRP 2932

Este bloque es indispensable para establecer la comunicación entre el NI USRP 2932 y la computadora que contiene el software de aplicación LabVIEW para el procesamiento de banda base de la señal.

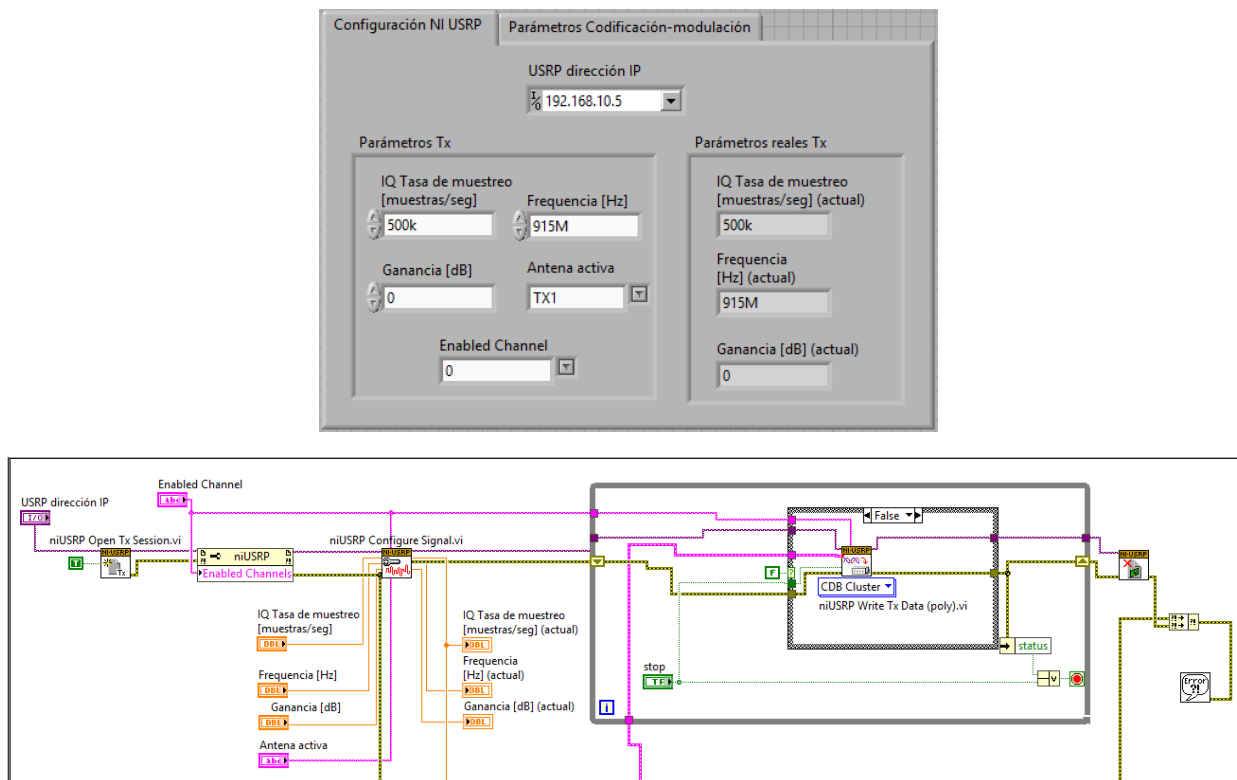


Figura 4.16: Parámetros de configuración transmisor NI USRP 2932.

En la figura 4.16 se muestran los parámetros configurables. Primeramente, se establece la dirección IP del dispositivo NI USRP con ayuda del bloque de inicio de sesión de transmisión (*niUSRP Open Tx Session.vi*). A través del bloque de configuración de señal (*niUSRP Configure Signal.vi*) se define la frecuencia de la portadora dentro del rango de operación del radio (400 MHz-4400 MHz), en este caso específicamente se fija en las 915 MHz (correspondiente a la banda de 902-928 MHz); la ganancia de la antena; se selecciona el puerto de antena activa (TX1, solo existe una antena transmisora, para recepción si existen dos puertos de antena); y finalmente uno de los parámetros más importantes la tasa de muestreo IQ expresada en muestras por segundo. Adicionalmente aparecen los valores reales de configuración (el NI USRP depende de la tarjeta *daughterboard*, por lo que valores fuera de rango no serán aceptados).

La última sección del bloque de configuración del NI USRP está enfocada a la transmisión del flujo de información. Hace uso de un clúster (*niUSRP Write Tx Data (poly).vi*) que almacena y transmite los datos desde el *software* de operación (LabVIEW) al SDR. Dado que el clúster se encuentra dentro de un ciclo, la transmisión se detendrá únicamente cuando el usuario detenga intencionalmente la ejecución o cuando dicho buffer de memoria llegue a su límite (en el último caso, se detiene la secuencia y se envía un mensaje de advertencia al usuario).

4.3.1.2. Bloque conversor de texto a bits

Su función es convertir la secuencia de texto que se desea codificar a datos binarios que serán utilizados como parámetro de entrada del codificador BPSK^{+S}. Se utiliza el instrumento virtual *StringtoBin.vi* como base de operación del bloque.

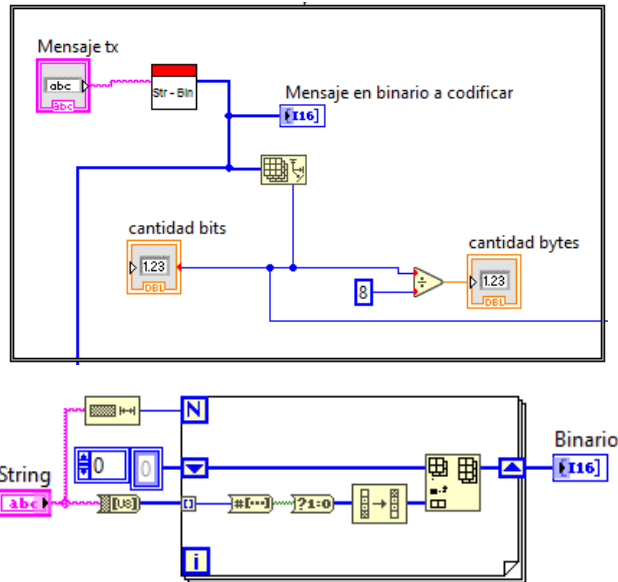


Figura 4.17: Conversor de texto a bits.

4.3.1.3. Bloque codificador BPSK^{+S}

Bloque encargado del proceso de codificación. Utiliza como parámetros de entrada la secuencia de bits a codificar, el tamaño de ventana de codificación, el tamaño de bloque, la cantidad de silencios seguidos permitidos y la opción de codificar o no. El tamaño de ventana de codificación es empleado por el codificador para definir que tanto reduce el número de bits a ser modulados, un tamaño de ventana de codificación $w=0$ representa un porcentaje de reducción de bits, nulo (contrario a lo deseado), pero evitaría la ocurrencia del efecto BERA; de ahí que a partir de mecanismos de detección de calidad de enlace para paquetes recibidos en el estándar IEEE 802.15.4, el transmisor podría establecer un tamaño de ventana cero para canales muy ruidosos. El tamaño de bloque es otra herramienta empleada en el minimizado del efecto BERA (aunque no lo elimina por completo), el segmentado de los datos a codificar en pequeños bloques provoca que al ocurrir un error en un bloque en específico este no compromete toda la trama de datos sino solo el bloque donde ocurre el error. La cantidad de silencios seguidos permitidos antes de perder la sincronización es un parámetro experimental, cuyo valor será determinado en base a la implementación realizada y a través de pruebas experimentales; es importante resaltar que este parámetro significa la necesidad de agregar un estado de relleno (estado «1») por cada x cantidad de silencios seguidos a transmitir. La opción de codificar o no, significa el empleo de la dualidad presente en la modulación BPSK^{+S}, que brinda la posibilidad que un modulador BPSK^{+S} pueda ser utilizado como modulador BPSK simple.

Se emplea el MATLAB *script node* para la programación mediante MATLAB del codificador (Apéndice A.2), a la salida se obtiene el mensaje codificado con silencios más una cabecera que se mantiene sin codificar para mayor robustez. Ver figura 4.18.

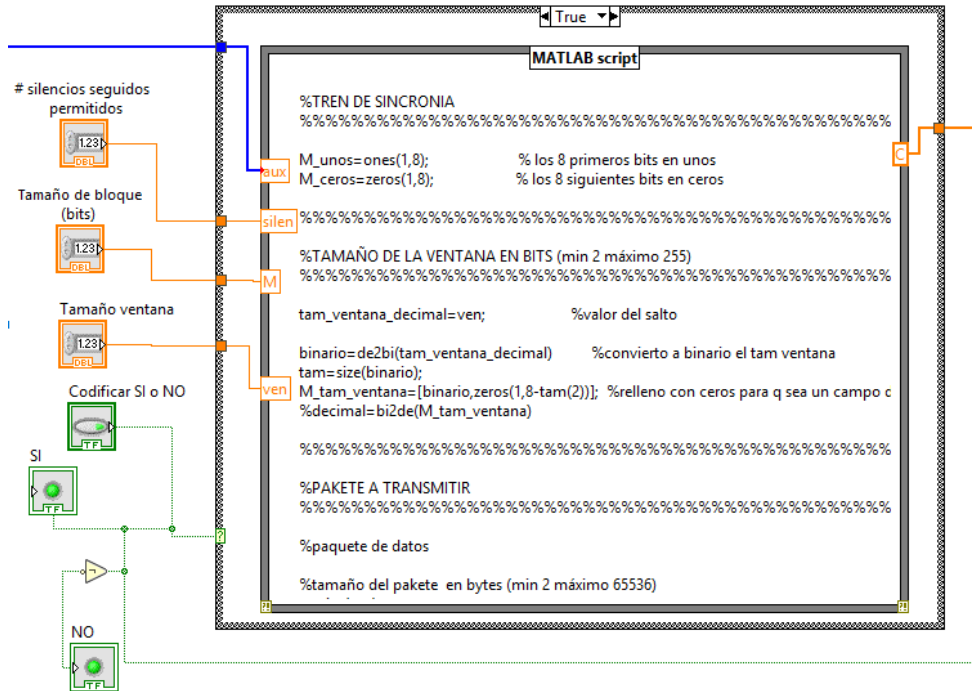


Figura 4.18: Codificador BPSK^{+S} en LabVIEW.

La estructura de la cabecera BPSK^{+S} (72 bits, 9 bytes) comprende los siguientes campos:

1. Preámbulo: 32 bits que se utilizan para sincronización de símbolos. Contiene los bits de guarda y los bits de sincronía.
2. Delimitador: 16 bits que se utilizan para sincronizar la recepción de la trama. Como bits de inicio de trama se utilizó el carácter “~ ~”.
3. Tamaño de ventana: 16 bits que especifican el tamaño de ventana de codificación.
4. Tamaño paquete: 8 bits que especifican la longitud de la unidad de datos en bytes (*payload*).

La tabla 4.2 muestra la estructura final del paquete de datos, tomando como ejemplo el texto: “*Esto es una prueba de la modulación BPSK más silencios*”.

Tabla 4.2: Estructura del paquete final a transmitir.

Campo	Número de bits	Estructura
Bits Guarda	12	Matriz unos y ceros
Bits Sincronía	20	Sec. pseudoaleatoria
Bits inicio trama	16	[0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0]
Bits tamaño ventana en bits	16	-
Bits tamaño paquete en bytes	8	-
Bits Mensaje	448	≤ 127 bytes
Bits Redundantes	8	[0 0 0 1 1 1 1 1]

La cadena de datos correspondientes al mensaje codificado en conjunto con los bits de guarda, bits de sincronía, bits de inicio de trama, bits de tamaño de ventana, bits de tamaño de paquete y bits redundantes forman el paquete final de información que posteriormente pasará al proceso de modulación para, finalmente, ser transmitido.

4.3.1.4. Bloque modulador BPSK^{+S}

Este bloque es el encargado del proceso de modulación de los datos ternarios recibidos. El modulador diseñado se compone de tres elementos principales: un modulador base BPSK preimplementado en LabVIEW a partir del *Modulation Toolkit*, el bloque *preparación-para-envio.vi* y el bloque *modulador-silencios.vi*. Ver figura 4.19.

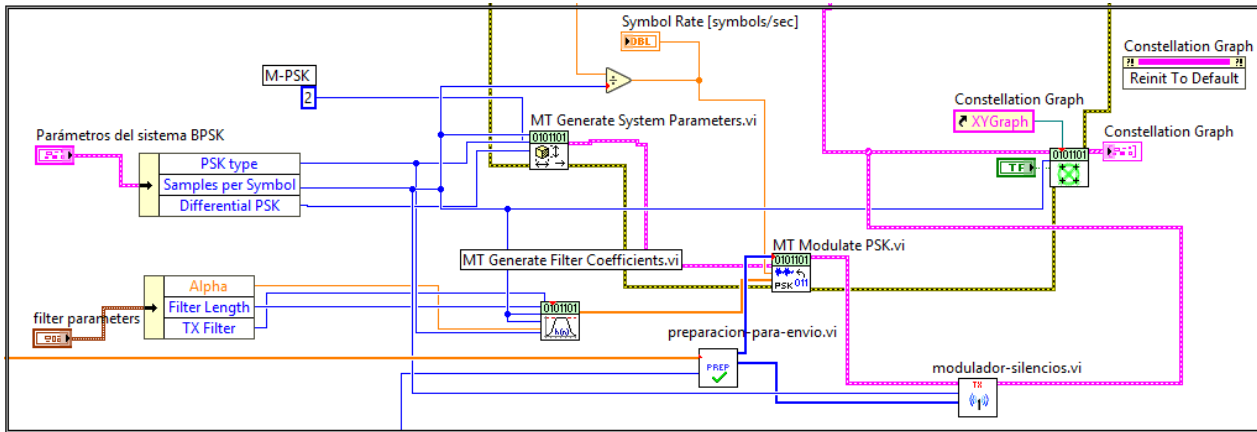


Figura 4.19: Modulador BPSK^{+S} en LabVIEW.

El bloque *preparación-para-envio.vi* se encarga de separar la información ternaria recibida del bloque codificador en dos grupos, el que contiene los estados uno y cero propios de la modulación BPSK y el del estado S, propio de la técnica de codificación-modulación BPSK^{+S}, se envía el primer grupo al bloque *MT Modulate PSK.vi* donde se efectúa la modulación BPSK estándar y el segundo grupo se manda al bloque *modulador-silencios.vi* para que luego de realizada la modulación estándar BPSK este bloque conforme los datos finalmente a enviar por el canal, modulados en BPSK^{+S}.

Resulta necesario la utilización de un filtro antes de la modulación, debido a que pese a ser una modulación simple resulta muy complicado para el demodulador recuperar los datos. El filtro de coseno elevado se emplea para minimizar la interferencia entre símbolos (ISI) y combatir los inconvenientes que pueda provocar en la recepción y demodulación de la señal. El bloque *MT Generate Filter Parameters.vi* valida los parámetros definidos por el usuario y genera un vector que contiene los coeficientes del filtro.

4.3.2. Receptor BPSK^{+S}

El receptor implementado al igual que el transmisor tiene cinco bloques principales: configuración del NI USRP 2932, conversor de bits a mensaje, demodulador BPSK^{+S}, decodificador BPSK^{+S} y presentación de datos. Ver figura 4.20.

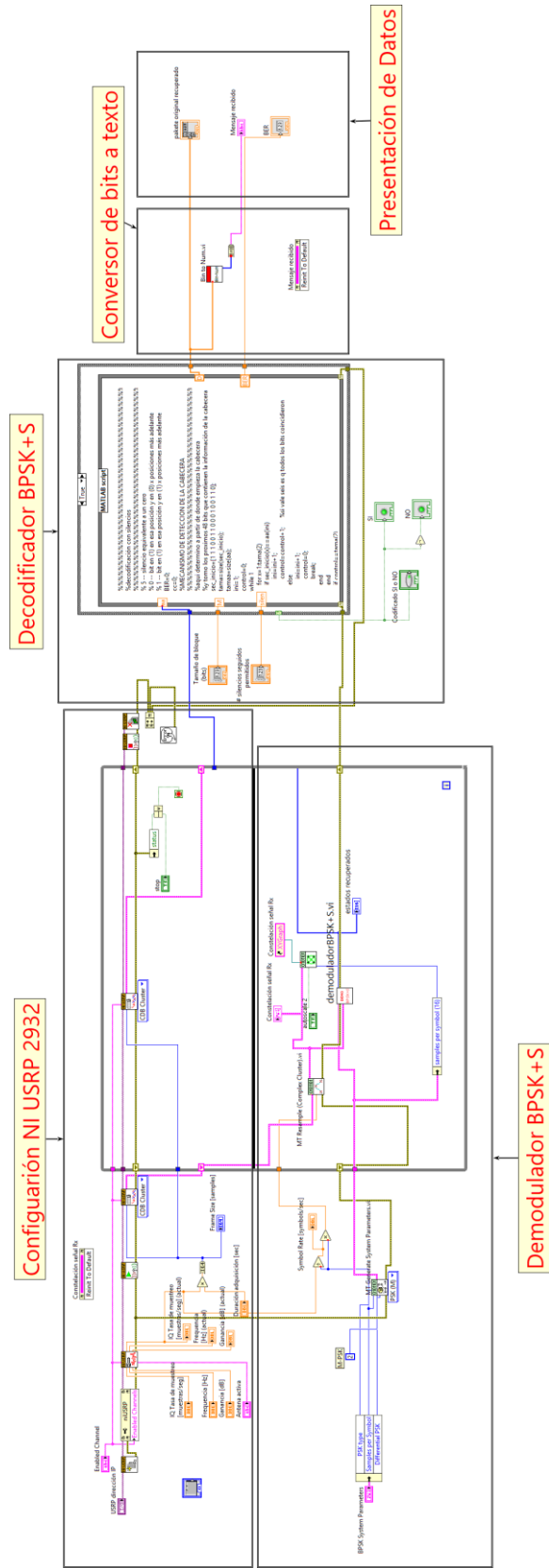


Figura 4.20: Diagrama en bloques receptor BPSK^{+S} en LabVIEW.

En la figura 4.21 se muestra el panel frontal del receptor. La parte superior izquierda representa los parámetros de configuración del NI USRP y los de demodulación- codificación BPSK⁺S. La parte superior derecha representa el diagrama de la constelación de recepción. En un espacio central se muestra el mensaje recibido (“*Esto es una prueba de la modulación BPSK más silencios*”) que demuestra la correcta operación del receptor. En la parte inferior se muestran el mensaje codificado recibido (salida del bloque demodulador BPSK⁺S); también aparece el mensaje decodificado y el BER.

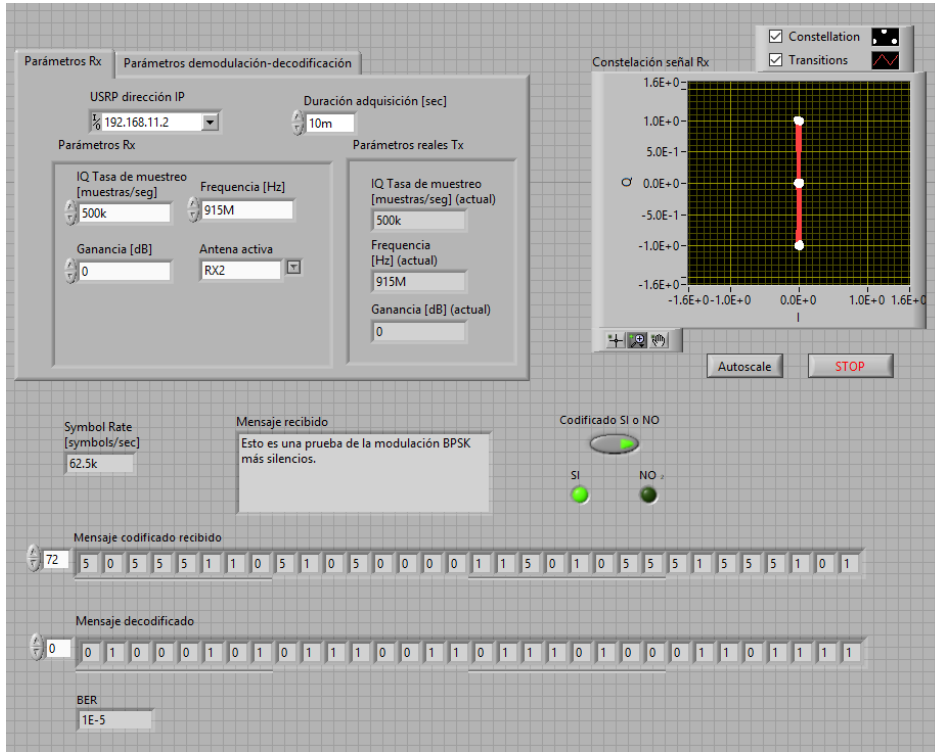


Figura 4.21: Panel Frontal receptor BPSK⁺S en LabVIEW.

El dispositivo de recepción es configurado con los mismos parámetros de configuración del NI USRP transmisor: tasa de muestreo y frecuencia de la portadora (deben ser idénticos para una correcta sincronización). Se define la dirección IP, la ganancia de la antena en recepción, el puerto de antena activo (el modelo NI USRP 2932 presenta dos puertos de antena en recepción). Ver figura 4.22.

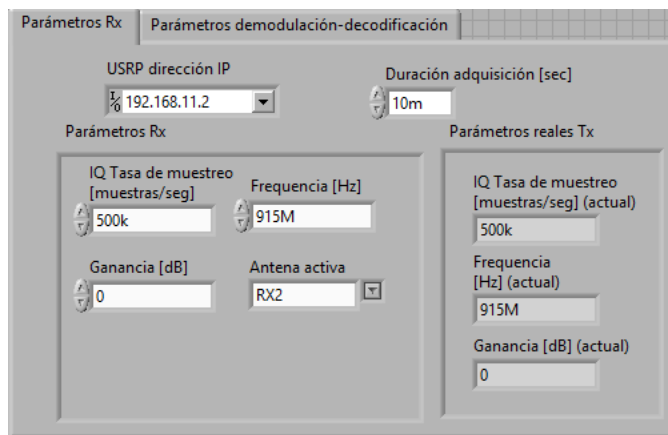


Figura 4.22: Parámetros de configuración receptor NI USRP 2932.

Mediante el módulo *niUSRP Fetch Rx Data (poly).vi* perteneciente a la sección de configuración del NI USRP, se obtienen los datos adquiridos por la antena, para que la función trabaje correctamente hay que introducirle, por dato, el número de muestras que se quiere obtener en la adquisición del canal. Para determinar un correcto tiempo de adquisición que permita recibir todas las muestras transmitidas que contienen la información de los estados BPSK^{+S} se sigue la siguiente relación:

$$\frac{\text{tasa de muestreo IQ}}{\text{muestras por símbolo}} = \frac{500\,000 \text{ muestras/seg}}{8 \text{ muestras/símbolo}} = 62\,500 \text{ símbolos/seg}$$

En el caso específico de la modulación implementada un símbolo representa un estado («0», «1» o S) transmitido. El tiempo de adquisición establecido para el ejemplo de la figura 4.21 fue de 10 milisegundos, lo que permite recibir 625 símbolos o estados.

Los valores complejos de cada símbolo serán enviados a los dos bloques principales de recepción: el *MT Resample (Complex Cluster).vi* y el demoduladorBPSK+S.vi.

El *MT Resample (Complex Cluster).vi* realiza una reducción lógica en el número de muestras tomadas de acuerdo a la entrada, que debe ser igual a la tasa de muestreo indicada en el transmisor (se realiza un nuevo proceso de muestreo). Dicho de forma más simple, este módulo se encarga de realizar una decimación virtual cuya salida sea una forma de onda compleja con una tasa de muestreo similar a la obtenida por el modulador BPSK^{+S}, en el transmisor.

El bloque demoduladorBPSK+S.vi. realiza el proceso de demodulación basado en los umbrales de decisión establecidos, obteniéndose a la salida los estados «0», «1» o S; que son enviados al bloque decodificador.

Para el proceso de decodificación es necesario conocer el tamaño de bloque, la cantidad de silencios seguidos permitidos y la opción de codificar o no, parámetros que son definidos en receptor y son los mismo definidos en el transmisor. El tamaño de ventana de codificación y el tamaño del paquete son campos contenidos en la cabecera BPSK^{+S}, el propio decodificador los extrae de la secuencia de datos recibida (Apéndice A.3). Una vez desarrollado correctamente el proceso de decodificación, los bits de salida del bloque son enviados al bloque conversor de bits a texto donde finalmente se genera el mensaje recibido.

Capítulo 5

Experimentos y resultados

En este capítulo se dará una descripción de los experimentos, así como de los resultados obtenidos y su análisis.

5.1. Metodología de los experimentos

Con el fin de comprobar la funcionabilidad de la nueva técnica de codificación-modulación BPSK^{+S}, en un primer momento los experimentos estarán enfocados a la comparación de la constelación de transmisión y de recepción, la codificación y decodificación correcta del mensaje, el estudio y análisis del BER teórico contra el BER obtenido en el funcionamiento real del sistema y por simulación a través de la agregación de un bloque de ruido AWGN, así como el número de silencios seguidos permitidos antes de verse afectada la sincronización en el proceso de recepción de los paquetes. En un segundo momento se realizarán simulaciones a través de una herramienta desarrollada en MATLAB para determinar los valores óptimos de tamaño de ventana y bloque de codificación, se muestran gráficas comparativas del número de estados a transmitir, y el promedio de silencios o estado S presentes en los mensajes codificados y posteriormente modulados con el consecuente ahorro energético en comparación con la modulación BPSK convencional, para condiciones de relación señal a ruido donde ambas modulaciones pueden ser aplicables y basados en la característica de que muchos nodos sensores por su simplicidad no cuentan con mecanismos de adaptación de potencia de transmisión (tienen una potencia fija).

En la tabla 5.1 se listan los experimentos realizados y su función.

Tabla 5.1: Objetivos de los experimentos a realizar.

Experimento	Objetivo
Selección del canal y SNR	Verificar que el canal donde se realicen los experimentos se encuentre libre de señales interferentes de otras fuentes y determinar los valores de potencia del ruido y potencia de transmisión para obtener la SNR.
Validación de la implementación	Comprobar el correcto funcionamiento de la implementación BPSK ^{+S} .
BER en BPSK ^{+S}	Comparar BER teórico, real y simulado.
Variación ventana de codificación	Estudio de la influencia de la variación de la ventana de codificación en el número de estados a modular.
Variación del tamaño de bloque	Estudio de la influencia de la variación del tamaño de bloque de codificación en la minimización del efecto BERA.
Número óptimo de silencios seguidos	Determinar el valor máximo de estados S o silencios seguidos permitidos antes de que se afecte la sincronización.

5.2. Selección del canal y SNR

El funcionamiento de la nueva técnica de codificación-modulación BPSK^{+S} implementada en los radios NI USRP 2932 es validado a partir de la realización de un experimento de transmisión-recepción de un paquete de datos.

El primer paso consiste en determinar una frecuencia de operación adecuada. Debido al rango de operación de 400 MHz a 4.4 GHz de los equipos NI USRP 2932 y las antenas disponibles, se decidió trabajar en la frecuencia 915 MHz con la antena monopolo VERT400. La banda de 902 MHz a los 928 MHz es utilizada por radioaficionados, es una banda relativamente libre, por lo que existen pocas señales interferentes que perjudiquen el correcto desempeño del sistema.

Para comprobar que la frecuencia en la que se efectuarían los experimentos estuviera libre se hizo uso de un solo NI USRP y el instrumento virtual (VI) *spectral monitoring (interactive).vi*, incluido en los ejemplos del NI USRP en LabVIEW. Este instrumento virtual está orientado al bosquejo del espectro de potencia de una señal variante en el tiempo. Al ser un instrumento de recepción, hace uso de los bloques *niUSRP configure signal.vi*, *niUSRP initiate.vi* y de su respectivo clúster de recepción. Dentro de los parámetros de configuración del SDR, el control *I/Q Sampling Rate* (que indica el número de muestras tomadas en el tiempo) define el número de elementos que analizará el bloque de la FFT (*fast fourier transform*); por decirlo de alguna forma, establece el "ancho de banda" del espectro de salida. El bloque se encarga de realizar un promedio estadístico de distintas FFT (10 como mínimo), a fin de ofrecer un arreglo final de datos que describan más cercanamente el monitoreo espectral (realiza la autocorrelación empleando un truncamiento por ventana rectangular). De esta forma se obtiene el valor conjugado de cada uno de los elementos obtenidos anteriormente y los multiplica con los elementos del primer arreglo (obtiene el cuadrado del valor promedio de voltaje recibido en un símbolo sobre una carga de 50 ohms). Finalmente, divide el producto de ambos entre el número de elementos analizados, obteniendo así, el espectro de potencia.

En la Figura 5.1 se puede notar un pico en la frecuencia de trabajo, este se debe al oscilador local del propio equipo. A excepción de ello, se aprecia la ausencia de señales en el ancho de banda del espectro donde se realizarán los experimentos.

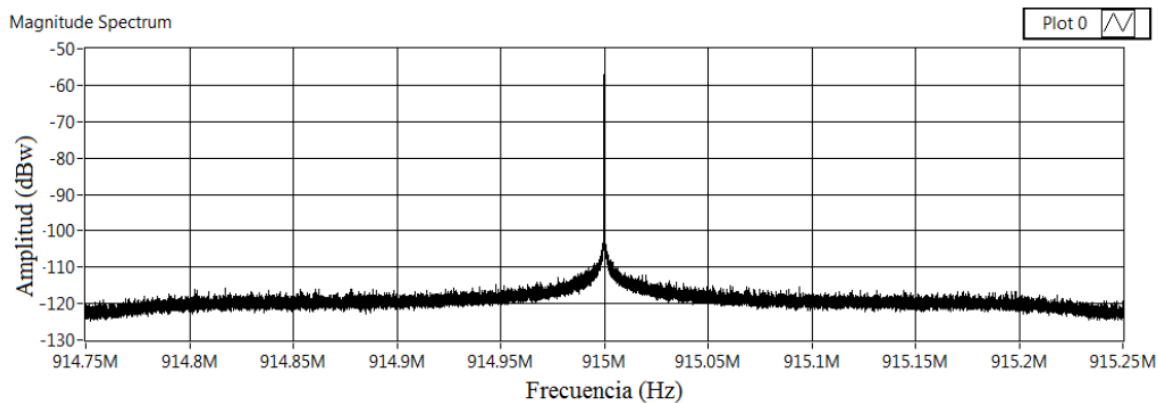


Figura 5.1: Espectro de potencia del canal libre.

Se hace necesario determinar la potencia del ruido y la potencia de transmisión para obtener los valores de SNR utilizados en la gráfica de la tasa de bit erróneos. Se procesó el espectro de potencia del canal libre de la figura 5.1 con ayuda de MATLAB y se determinó que la potencia promedio del ruido es de -117.810 dBW o -87.810 dBm.

Para lograr una variación en la relación señal a ruido se varió la ganancia del transmisor dejando fija la distancia entre transmisor y receptor (3 metros). Los dispositivos NI USRP 2932 como se explicó en el epígrafe 4.1.1, admiten un rango ganancia de 0 a 31.5 dB, para tener más muestras se hizo uso de un atenuador de 30 dB. Al variar la potencia del nodo transmisor a través de la ganancia durante la realización de los experimentos, se notó como variaba la amplitud en el espectro que se recibió en el nodo receptor como se muestra en la figura 5.2, con estos espectros y ayuda de MATLAB se obtuvo la potencia promedio de cada uno en el ancho de banda en que se realizaron los experimentos. La tabla 5.2 muestra los valores de potencia de transmisión obtenidos y la SNR.

Tabla 5.2: Niveles de potencia de transmisión y SNR.

Ganancia transmisor	dBW	dBm	SNR
+30	-99.246	-69.246	18.56
+20	-102.053	-72.053	15.76
+10	-105.162	-75.162	12.65
+0	-108.449	-78.449	9.36
-10	-113.530	-83.530	4.28
-20	-115.878	-85.878	1.93
-30	-117.804	-87.870	0

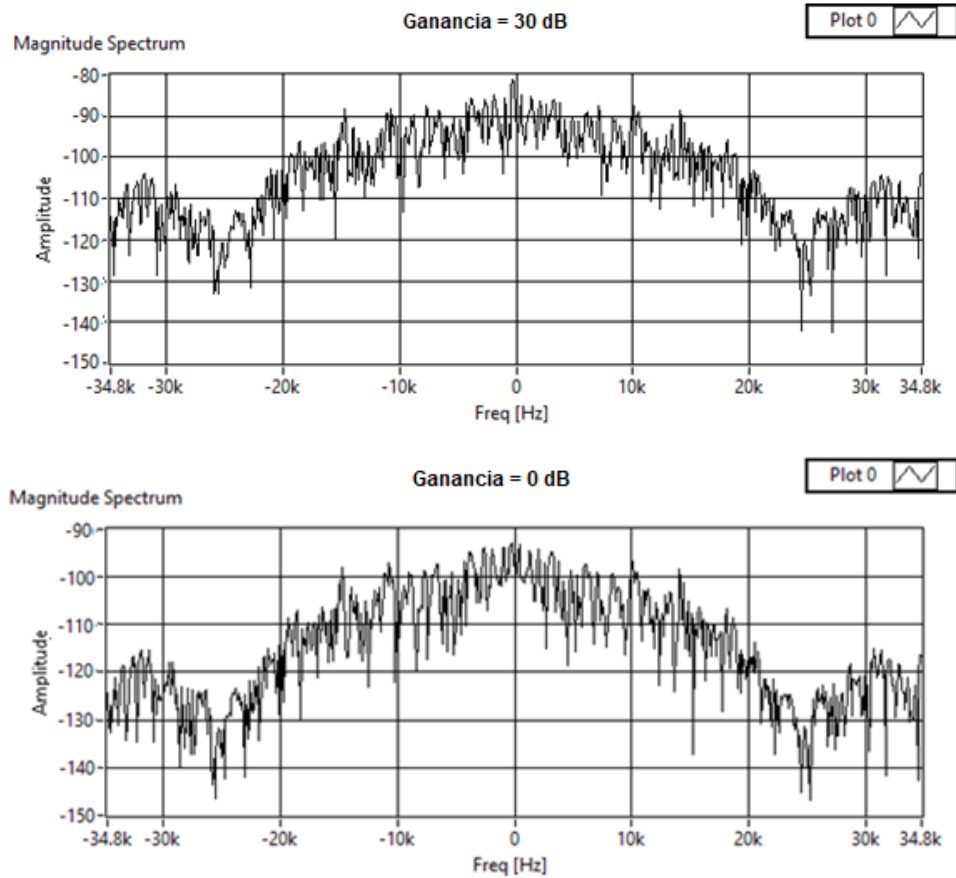


Figura 5.2: Amplitud de la potencia espectral.

5.3. Validación de la implementación de BPSK^{+S}

El experimento se realizó en un espacio cerrado, donde los dispositivos (transmisor y receptor BPSK^{+S}) tenían una línea de vista con aproximadamente 3 metros de distancia máxima entre ellos.

El próximo paso consiste en configurar los parámetros de transmisión y recepción. Para el caso del transmisor se define la dirección IP 192.168.10.5, la frecuencia de operación en 915 MHz, la tasa de muestreo IQ de 500 mil muestras por segundos a razón de 8 muestras por símbolo, que da como resultado 62 500 símbolos por segundo. En cuanto a los parámetros de codificación-modulación se establece un tamaño de ventana $w=1$, un tamaño de bloque de codificación $m=24 \text{ bits}$, el número de silencios seguidos permitidos se define en 3, además de los coeficientes del filtro *root raised cosine*. En el lado receptor todos los valores deben coincidir a excepción de la dirección IP que en este caso sería 192.168.11.2 y la duración de la adquisición (10 ms), se omite el tamaño de ventana por venir contenida en la cabecera BPSK^{+S}. Los valores de tamaño de bloque de codificación y número de silencios seguidos solo son utilizados con fines experimentales que permitan determinar su valor óptimo, en consecuencia, una vez conocidos estos valores, pasarán a ser parámetros fijos sin necesidad de ser definidos.

La figura 5.3 se observa el diagrama de la constelación en transmisión y recepción con el sistema en funcionamiento. Se encuentran bien definidos los tres estados modulados, en los extremos superior e inferior los estados «0» y «1» respectivamente, con su diferencia de fase de 180 grados, y en el centro el estado S representando los silencios.

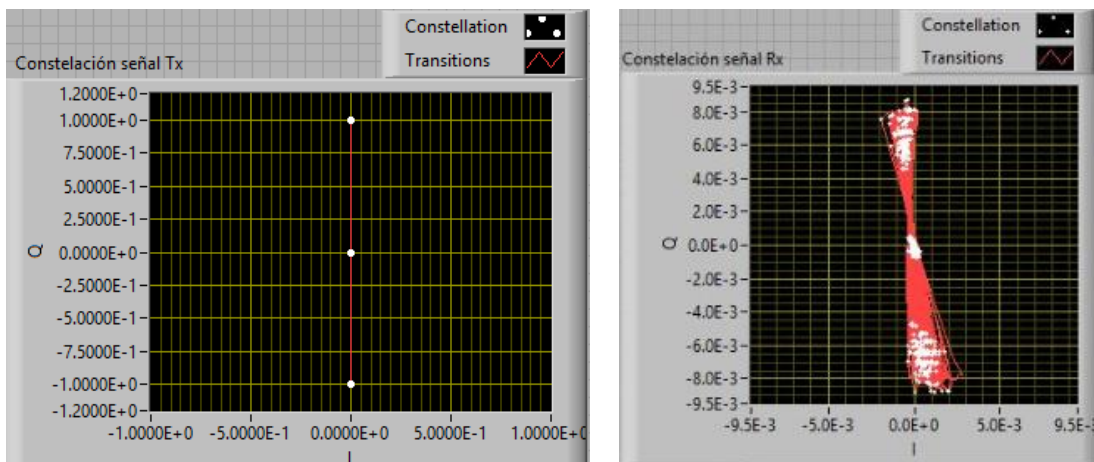
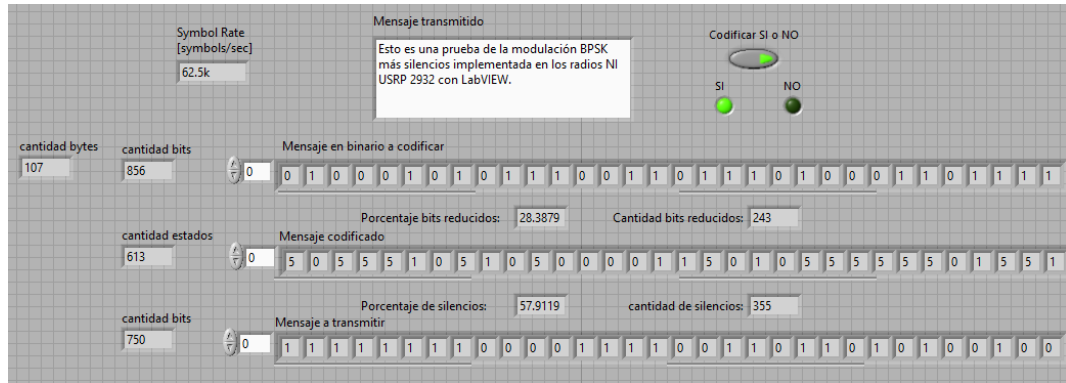


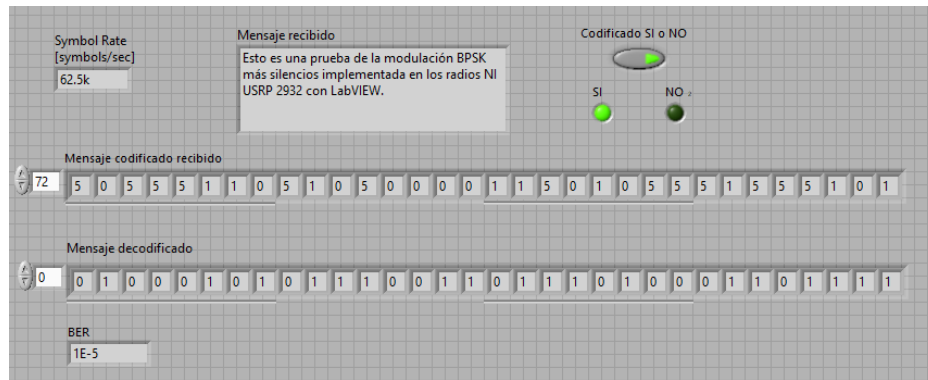
Figura 5.3: Constelación de BPSK^{+S} en transmisión y recepción.

Se observa la dispersión de los puntos de la constelación de recepción con respecto a la constelación ideal que puede deberse al ruido de fase del canal.

A continuación, se transmite la cadena de caracteres: “*Esto es una prueba de la modulación BPSK más silencios implementada en los radios NI USRP 2932 con LabVIEW.*” que contiene 856 bits (107 bytes) en código ASCII. Para los valores de tamaño de ventana y bloque de codificación antes definidos, se obtiene una disminución en los bits luego de la codificación de un 28.38 % quedando en 613 la cantidad de estados; una vez agregada la cabecera (valor fijo de 9 bytes) y los bits de relleno por motivos de sincronización (evitar muchos silencios seguidos), el mensaje final a transmitir queda en 750 estados. Además, la cantidad de silencios y por ende espacios de ahorro energético es de 355 para un 47.33 % del total de estados a transmitir. Ver Figura 5.4.



a)



b)

Figura 5.4: Mensaje codificado y decodificado en BPSK^{+S}: a) en transmisión y b) en recepción.

Una vez validado el correcto funcionamiento de la implementación realizada, es necesario caracterizar parámetros importantes como la relación BER contra SNR.

5.4. BER en BPSK^{+S}

Con el fin de hacer una correcta comparación entre la tasa de error de bit determinada teóricamente en el epígrafe 3.4.3 y los valores reales obtenidos durante la implementación; se estableció una correspondencia entre cada valor de SNR de la tabla 5.2 y una medición del BER real con los radios en funcionamiento. Mediante un script de MATLAB implementado en LabVIEW se realizó una comparación estado a estado entre la trama transmitida y la recibida, para arrojar la relación entre el número de estados en error y el número total de estados transmitidos. La tabla 5.3 muestra los valores de BER contra SNR obtenidos en el experimento, se varió la ventana de codificación: valor de ventana cero y valor de ventana diferente de cero ($w = 1$). Se utilizó una secuencia de datos pseudoaleatorios con probabilidad 0.5 de ocurrencia de un uno o un cero en la secuencia binaria a codificar.

Tabla 5.3: Valores de BER transmisión real.

SNR	BER ($w = 0$)	BER ($w = 1$)
0	0.598990	0.608479
1.93	0.342712	0.369327
4.28	0.0769890	0.0773067
9.36	0.00221256	0.00498753
12.65	1.2526238e-03	2.2526238e-03
15.76	1.03256700e-04	3.43026238e-04
18.56	1E-06	1 E-06 ⁽⁴⁾

⁴ El valor del BER cuando tiende a cero se considera 1×10^{-6} para razones de graficado.

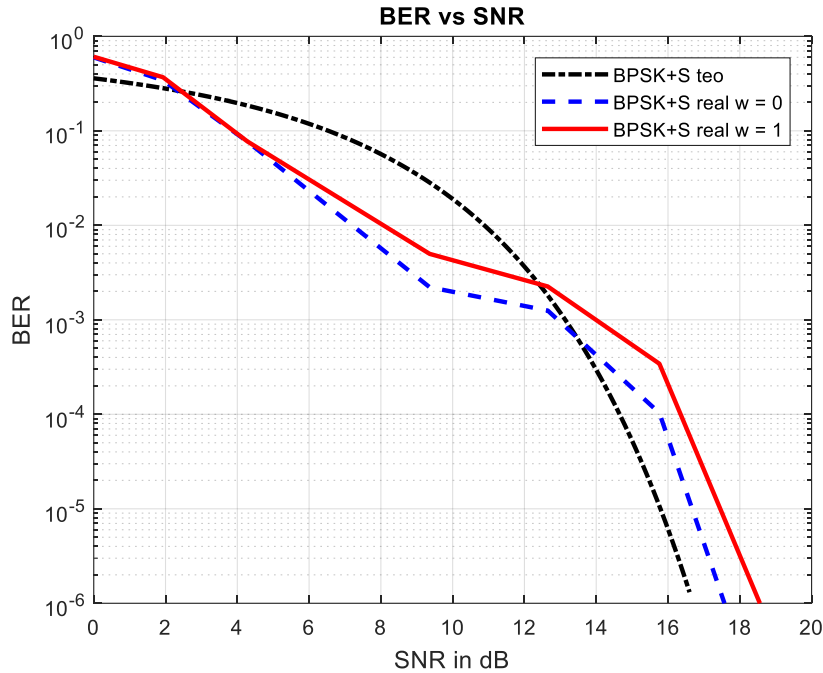


Figura 5.5: BER contra SNR en BPSK^{+S} teórico y real.

Una comparación más específica resulta de tomar los VIs desarrollados para el transmisor y el receptor BPSK^{+S} (validado su funcionamiento anteriormente) y conectar a la salida del transmisor un bloque de agregación de ruido blanco aditivo gaussiano (AWGN) (Figura 5.6); y finalmente la salida de este bloque a la entrada del receptor. Se toma la decisión de utilizar un bloque AWGN, debido a que trabajos anteriores con radios NI USRP: [65] y [71], refieren la dificultad de una correcta caracterización de la relación BER contra SNR cuando se realiza la simulación sobre el canal real; dada la complejidad para determinar la potencia de ruido y la potencia de transmisión adecuadamente.

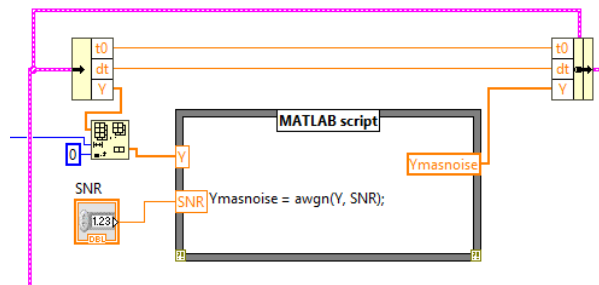


Figura 5.6: Bloque de AWGN.

La tabla 5.4 muestra los valores de BER obtenidos luego de variar la relación señal a ruido en el rango de 0-16 dB en el bloque de AWGN (es importante resaltar que se tomo el promedio de 10 mediciones en cada caso).

Tabla 5.4: Valores de BER transmisión simulada.

SNR	BER ($w = 0$)	BER ($w = 1$)
0	0.607201	0.608479
1	0.560951	0.561097
2	0.523522	0.531172
3	0.511765	0.511222
4	0.421443	0.421446
5	0.321235	0.334165
6	0.258823	0.269327
7	0.203499	0.226933
8	0.13992	0.15212
9	0.0531345	0.0773067
10	0.0251677	0.0374065
11	0.005341	0.0137506
12	0.0015121	0.0049875
13	0.0003112	0.0022656
14	5.00E-05	2.2526238 E-04
15	1.5E-06	3.4302623 E-05
16	1 E-6	1 E-6 ⁽⁵⁾

En la figura 5.7 se puede observar el comportamiento del BER con respecto a la SNR para el caso teórico y el simulado. Cuando el tamaño de ventana de codificación es igual a cero ($w = 0$) se aprecia una mejora en los valores de error de bit debido a que la codificación sería uno a uno (un estado «0» o «1» representa un solo bit codificado); si el valor de la ventana difiere de cero ($w = 1$) la codificación ya no sería uno a uno pues con un estado «0» o «1» se representan dos bits, las perturbaciones del canal de comunicaciones provocarían mayores errores en la recepción. En todos los casos se obtiene una SNR de hasta 16 dB aproximadamente con un error de bit mínimo de cero. La tasa de errores se eleva cuando la relación de la potencia de la señal con respecto al ruido comienza a bajar.

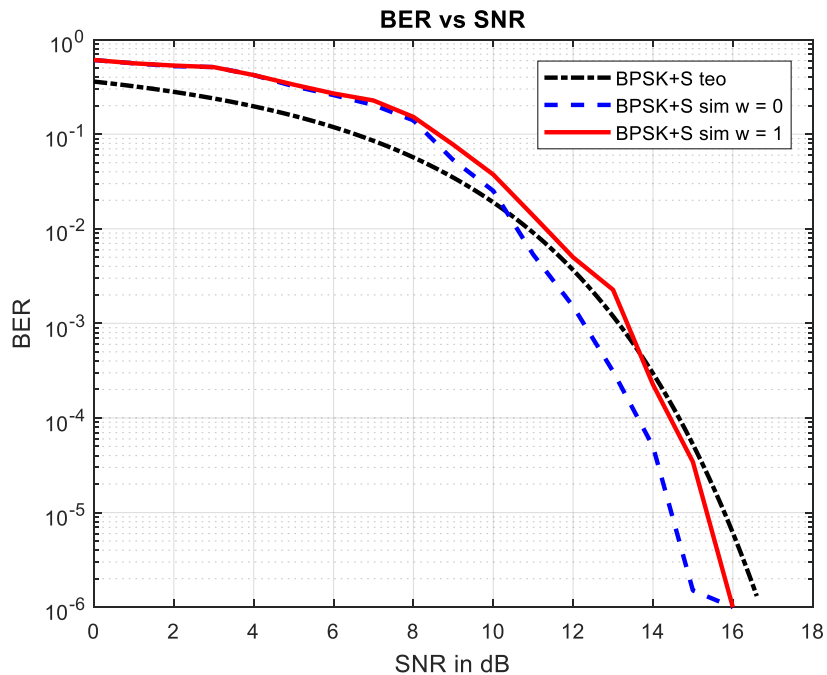


Figura 5.7: BER contra SNR en BPSK^{+S} teórico y simulado.

⁵ El valor del BER cuando tiende a cero se considera 1×10^{-6} para razones de graficado.

5.5. Variación ventana de codificación

En el epígrafe 3.4.1 se trató la introducción del concepto, tamaño de ventana, y su empleo en la codificación BPSK^{+S}. Para determinar el tamaño óptimo de ventana de codificación se desarrolló un programa en MATLAB (con auxilio del elaborado para el codificador BPSK^{+S}), a través del cual se varió el valor de la ventana desde cero hasta el total de bits a codificar. El programa acepta un texto de entrada en formato .txt, este es convertido a binario y codificado. Se fijó el tamaño de bloque de codificación igual al tamaño del paquete.

La figura 5.8 representa el comportamiento observado para la secuencia de ejemplo: “*Esto es una prueba de la modulación BPSK más silencios implementada en los radios NI USRP 2932 con LabVIEW.*”, que contiene 856 bits, a estos se adicionan los 72 bits de cabecera y los bits de relleno producto de la sincronización (83), se tomó un tamaño de bloque, $m = 856$ bits.

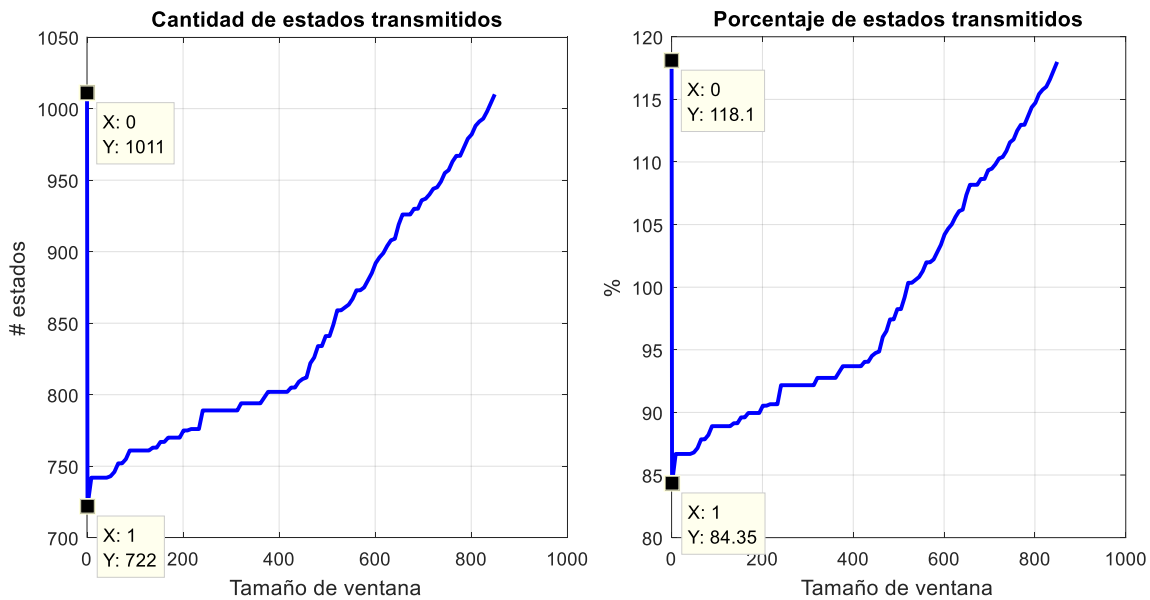


Figura 5.8: Relación tamaño de ventana de codificación y número de estados transmitidos totales.

Se observa que para una ventana de codificación $w = 1$, se logra la mayor reducción, pues solo se modularían 722 estados lo que representa el 84.35 %. Es importante destacar que para valores de tamaño de ventana mayores la mitad del número de bits a codificar se alcanzan valores superiores al 100 %, una consecuencia directa de la disminución del número de estados que codifican dos bits, además del agregado adicional que representa la cabecera y los estados de relleno.

Los estados finales modulados y transmitidos que representan gasto energético (estados «0» y «1») se observan en la figura 5.9. Se transmiten 382 estados para un 44.63 % de los bits iniciales a codificar.

La figura 5.10 muestra el porcentaje que representan estos estados («0» y «1») de los estados transmitidos totales. Todos son valores cercanos al 50 % es decir mitad estados («0» y «1») que representan gasto energético y mitad estados S o silencios.

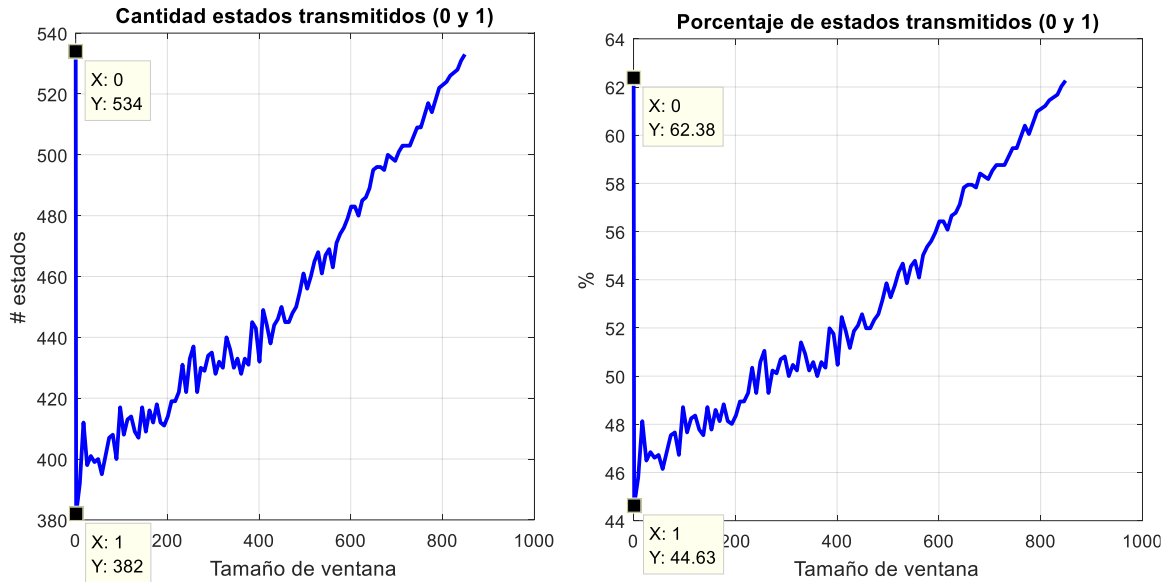


Figura 5.9: Relación tamaño de ventana de codificación y número de estados transmitidos (0 y 1) respecto al total de bits de la secuencia a codificar (856 bits).

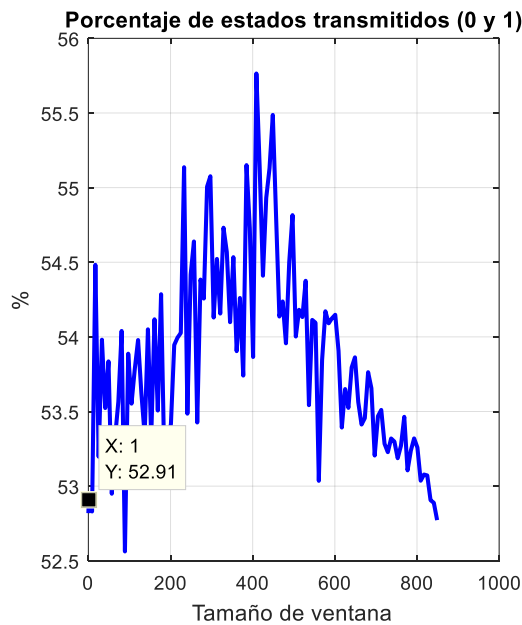


Figura 5.10: Relación tamaño de ventana de codificación y porcentaje de estados transmitidos (0 y 1) respecto al total de estados transmitidos.

No es posible dar un valor óptimo de ventana de codificación sin antes analizar el comportamiento de la tasa de errores con la variación de la ventana. En la figura 5.11 se observa como para valores de ventana $w = 1$ el BER es alto en comparación con la ventana $w = 0$ y valores alejados de uno, debido a que, si bien es cierto que para la ventana de valor uno se produce la mayor reducción de estados a transmitir, la codificación es uno a dos (con un estado se codifican dos) provocando que un simple error afecte dos estados. A medida que el tamaño de ventana aumenta la tendencia es disminuir la tasa de errores. Sin embargo, como los 16 dB es el límite mínimo de relación señal a ruido necesaria para un funcionamiento adecuado de la modulación BPSK^{+S}, es posible tomar como valor óptimo la ventana de valor uno.

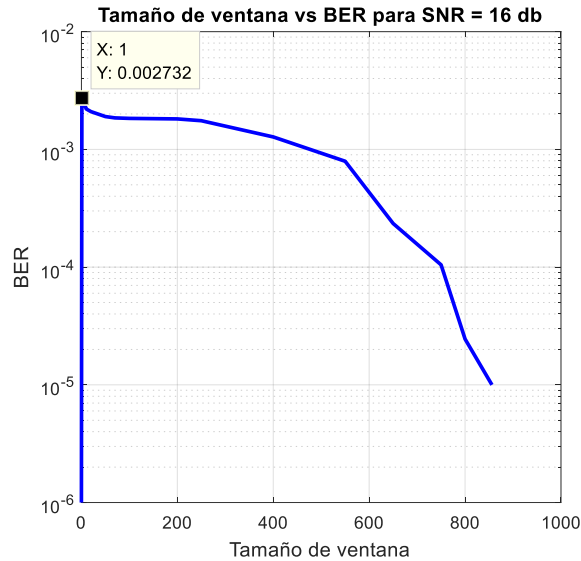


Figura 5.11: Relación tamaño de ventana de codificación y tasa de errores para SNR=16 dB.

El comportamiento promedio, luego de variar la ventana de codificación, el tamaño de la secuencia de datos a codificar y su composición (se tomaron varias secuencias de datos: secuencias pseudoaleatorias de probabilidad 0.5 de ocurrencia de un uno o un cero y de tamaños variables), sigue la regla del ejemplo anterior, siendo el tamaño óptimo de ventana $w=1$, con una reducción aproximada del 20 % (figura 5.12a) de los estados finales a modular y con un 50 % (figura 5.12b) de estados S o silencio en la secuencia final transmitida, con su consecuente ahorro energético.

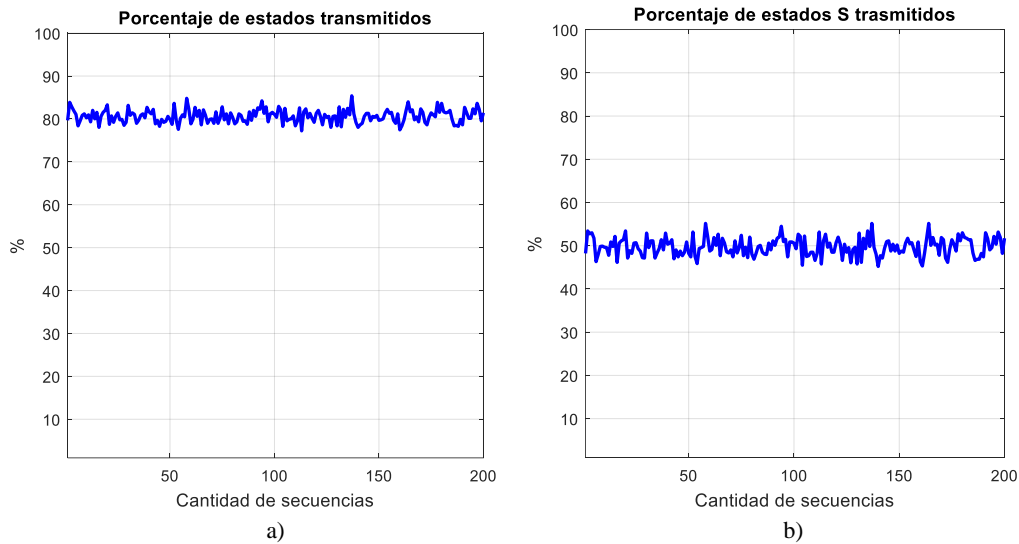


Figura 5.12: Porcentaje de estados transmitidos para ventana $w=1$: a) totales b) estados S.

Para el caso de canales muy ruidosos donde para el tamaño de ventana óptimo el BER siga siendo muy alto, se establecería un valor superior a uno y siempre menor al 50 % del total de bits del paquete a codificar; de persistir la pérdida de paquetes se establecería el valor mínimo de ventana ($w = 0$), donde la codificación sería uno a uno. El receptor a partir de la propiedad de la capa física de determinar el factor de calidad del enlace, y producto de que el transmisor no estaría recibiendo ACK de confirmación de recepción, fijaría el tamaño de ventana mínimo. Incluso en estas condiciones el número de estados S representa el 35 % del total de estados transmitidos, en promedio.

5.6. Variación del tamaño de bloque de codificación

El tamaño del bloque de codificación como se ha explicado en secciones anteriores, juega un papel determinante en disminuir el efecto de error de bit en avalancha (BERA). Es importante definir un valor adecuado de tamaño de bloque. Utilizando el mismo programa de determinación del tamaño óptimo de ventana, solo que manteniendo fijo en $w=1$ la ventana y variando el tamaño de bloque en el rango de uno hasta el número total de bits a codificar, para el mismo ejemplo de la sección anterior se obtienen los resultados de la figura 5.13.

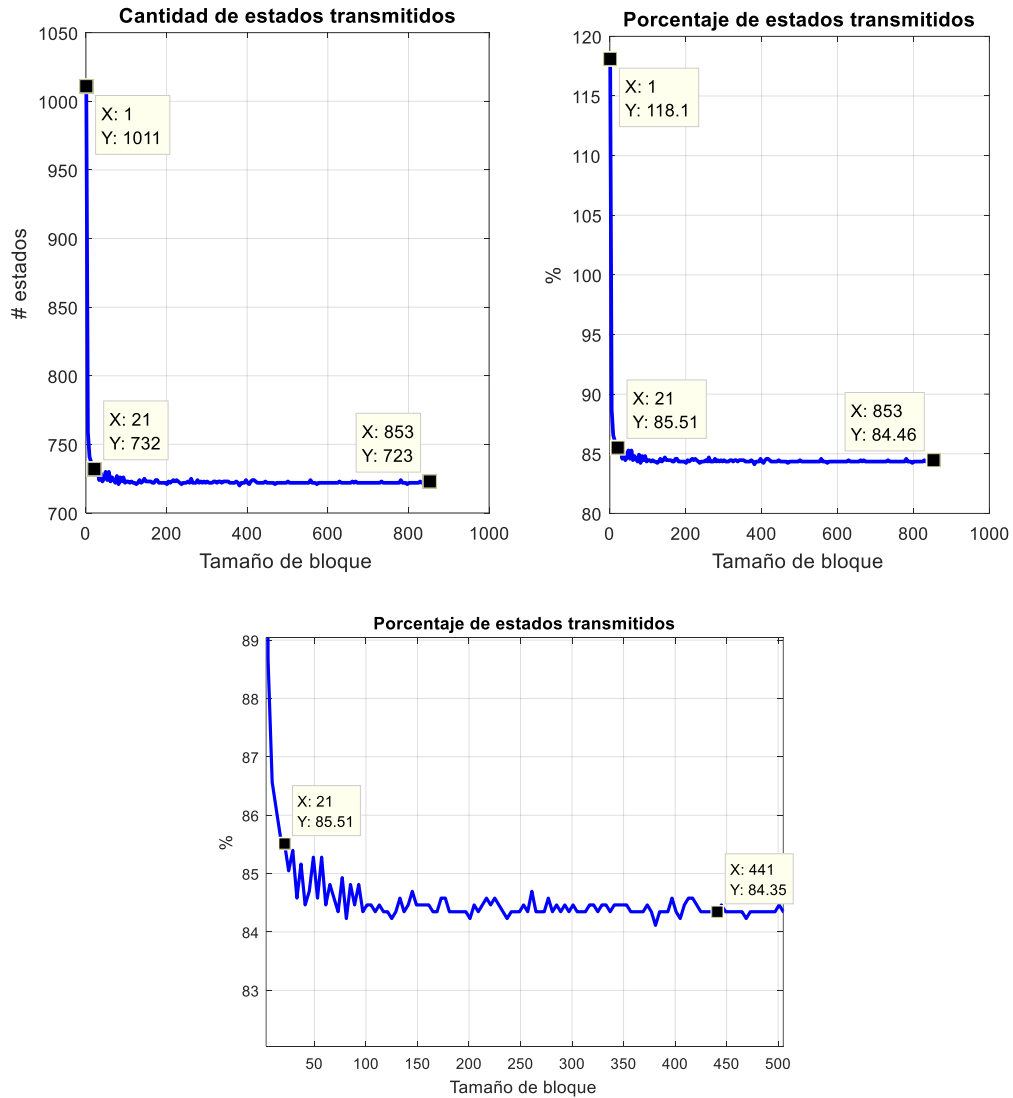


Figura 5.13: Relación tamaño de bloque de codificación y número de estados transmitidos totales.

Existe una relación inversa entre el tamaño de bloque de codificación y el número de estados codificados, a medida que aumenta el tamaño de bloque disminuye el número de estados hasta tender a un valor constante rápidamente. Para valores superiores a $m = 21$, la cantidad de estados en el ejemplo, tiende a 723 lo que representa un 84.46 %.

La figura 5.14 muestra la relación directa que existe entre el tamaño de bloque y el ahorro energético, pues un aumento del primero provoca una disminución del número de estados «0» y «1» modulados y en consecuencia menos tiempo en los modos de transmisión y recepción de los nodos sensores. Para valores superiores a $m = 21$,

la cantidad de estados que representan gasto energético tiende a 382 lo que representa un 44.63 % del total de bits iniciales antes de la codificación.

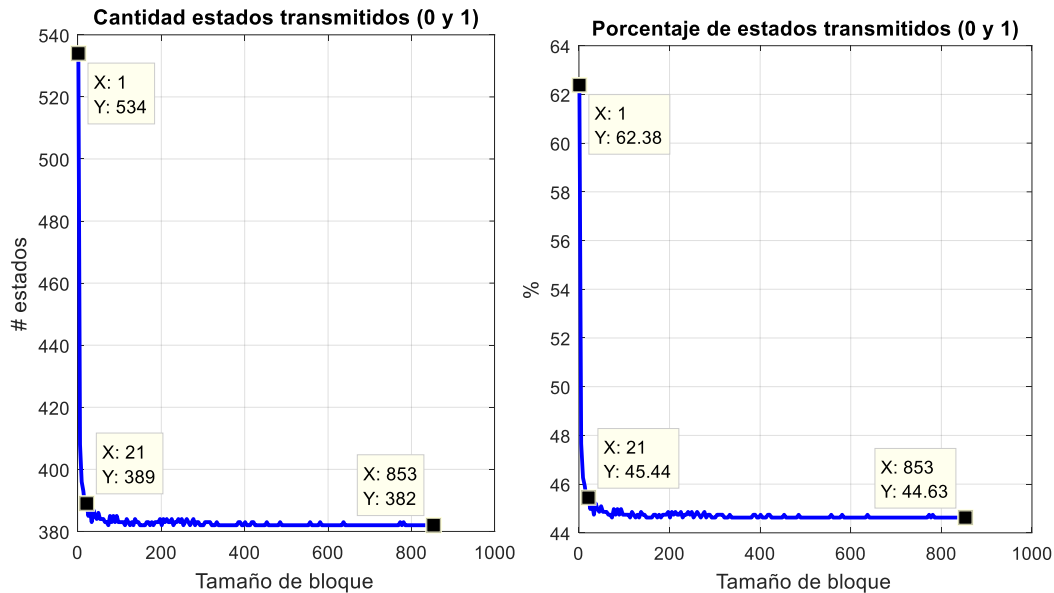


Figura 5.14: Relación tamaño de bloque de codificación y número de estados transmitidos (0 y 1).

El comportamiento promedio, luego de variar el tamaño del bloque de codificación, el tamaño de la secuencia de datos a codificar y su composición (se tomaron varias secuencias de datos: secuencias pseudoaleatorias de probabilidad 0.5 de ocurrencia de un uno o un cero y de tamaños variables), sigue la regla del ejemplo anterior, siendo el tamaño mínimo de bloque un valor superior a $m=24$ en promedio (figura 5.15a), con un 48 % (figura 5.15b) de estados «0» y «1» en la secuencia final transmitida.

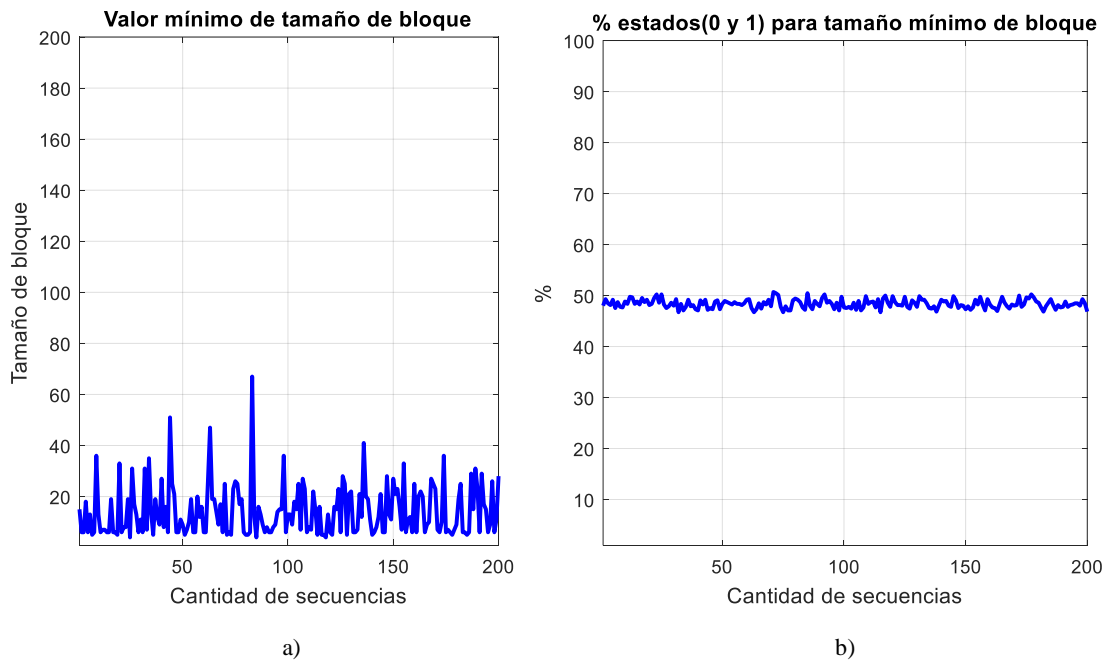


Figura 5.15: a) Número mínimo de tamaño de bloque y b) porcentaje de estados (0 y 1) transmitidos por cada secuencia.

Pudiera asumirse erróneamente que el valor ideal de bloque de codificación es igual a la cantidad máxima de bits a codificar, pero hay un factor que condiciona este parámetro: el efecto BERA.

5.6.1. Tamaño de bloque y efecto BERA

El siguiente experimento consiste en observar la influencia del tamaño de bloque en la minimización del efecto BERA. Se codifica la secuencia ejemplo (*Esto es una prueba de la modulación BPSK más silencios implementada en los radios NI USRP 2932 con LabVIEW.*) a partir del codificador implementado en MATLAB, el cual genera un archivo *.txt* con los estados codificados (*estados codificados.txt*), el decodificador recibe esta archivo y lleva a cabo el proceso de decodificación.

Con el fin de simular el ruido en el canal y los cambios que provoca en los estados transmitidos, se cambiaron manualmente algunos valores de estados por otros: estado S por estado cero o uno, estado cero por estado uno y estado cero o uno por estado S. El experimento se realizó para un tamaño de ventana $w=1$ y número de silencios seguidos permitidos tres, el tamaño de bloque m se fue variando hasta obtener un valor adecuado. Ver figura 5.16⁶.

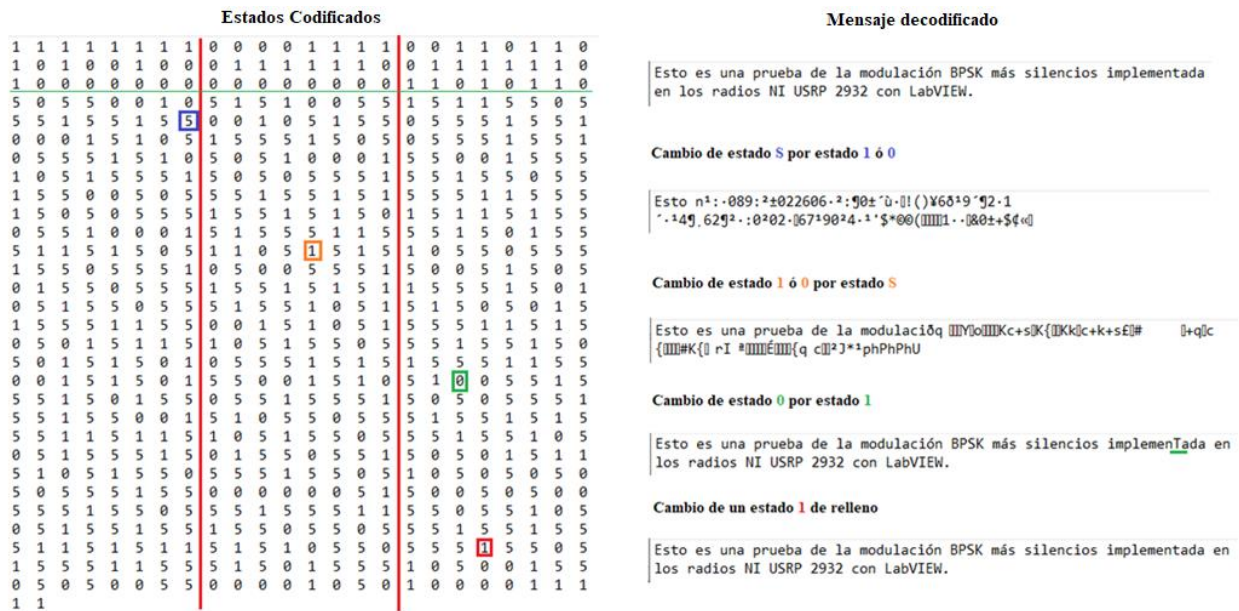


Figura 5.16: Efectos de los errores en el canal con $m=856$.

Se observa que un cambio de estado «0» por estado «1» solo afecta a una *bit* de la secuencia, el error provocado fue el cambio de una (*t*) minúscula por una (*T*) mayúscula, la secuencia 01110100 por 01010100 en ASCII. Si el cambio se produce en el estado «1» que se utiliza como relleno luego de varios silencios seguidos, este no afecta para nada los datos decodificados, pues el decodificador siempre desecha el estado que recibe luego de x silencios seguidos sin importar su valor. El caso más crítico lo representa un cambio de estado S por cualquier otro estado o viceversa, pues se desencadena el efecto BERA (la secuencia de datos queda irrecuperable, es necesaria la retransmisión completa). Es aquí donde el tamaño de bloque juega un papel determinante en la contención de la propagación de errores pues cada bloque se codifica independiente a pesar de mandarse concatenado en la trama completa de máximo 127 bytes.

La figura 5.17 representa el efecto de un tamaño de bloque de codificación adecuado; para $m=24$, los errores son contenidos y afectan solo a pequeñas partes de la secuencia de datos, errores que luego pueden ser corregidos en capa MAC con algún código FEC (*forward error correction*).

⁶ El número 5 en el archivo *estados codificados.txt* representa el estado S.

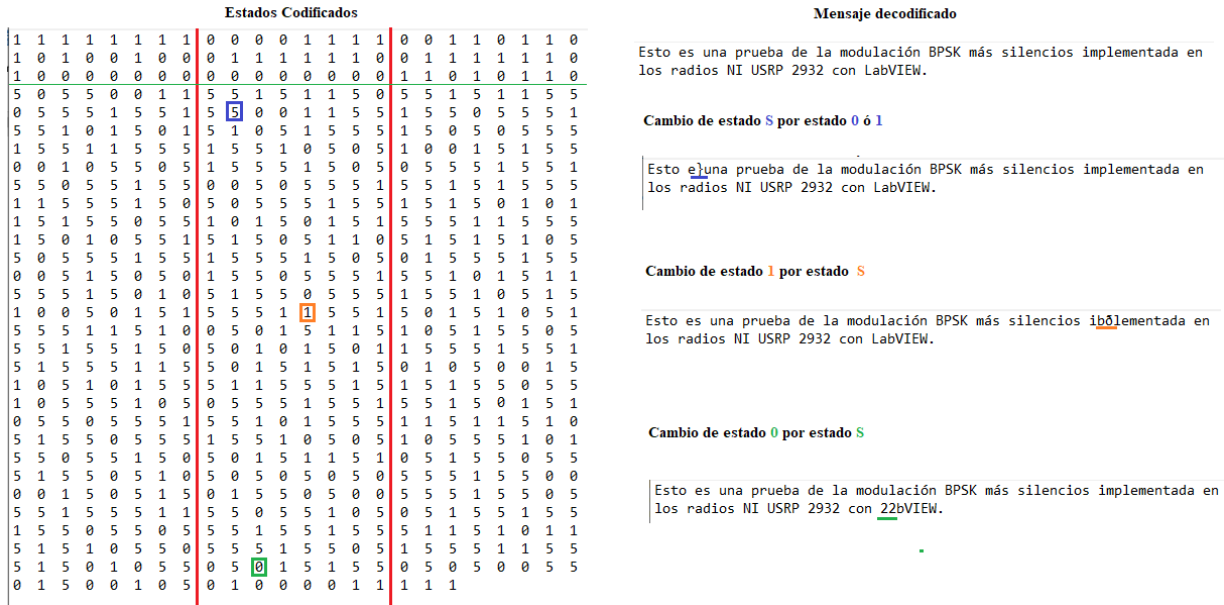


Figura 5.17: Efectos de los errores en el canal con $m=24$.

El experimento anterior fue repetido para diferentes secuencias de datos con composiciones de bits variadas y los resultados permitieron establecer como conclusión general que el tamaño óptimo de bloque de codificación es $m=24$, o valores cercanos. Pues a medida que aumenta este valor el número de errores crece exponencialmente y obligaría a una retransmisión de la trama completa. Ver tabla 5.5.

Tabla 5.5: Valores de BER contra tamaño de bloque para SNR=16 dB.

Tamaño de bloque	BER
1	1 E-6
5	1 E-6
10	1 E-6
20	1 E-6
24	1 E-6
40	1.5E-06
100	0.003112
200	0.258823
600	0.511765
856	0.607201

5.7. Número de estados S o silencios seguidos permitidos

El número máximo de silencios seguidos permitidos es un valor determinado experimentalmente. En la implementación realizada tanto en el transmisor como el receptor, se estableció el campo número de silencios para indicar el valor máximo luego del cual se agregaba el estado de relleno. Se realizaron múltiples envíos de secuencias de datos, entre transmisor y receptor utilizando los radios y se pudo observar que luego de tres silencios seguidos la sincronización comenzaba a comprometerse y era prácticamente imposible del lado receptor recuperar el mensaje transmitido. En consecuencia el número óptimo de silencios seguidos permitidos es tres.

La figura 5.18 expone la cantidad de estados de relleno agregados a una secuencia de datos codificados de 856 bits de tamaño, con un número de silencios seguidos permitidos de valor tres. Se destaca que para el tamaño de ventana óptimo $w=1$, el porcentaje de estados de relleno agregados no supera el 6 % y en el peor de los casos para canales muy ruidosos donde sea necesario establecer el tamaño de ventana mínimo $w=0$, en promedio los estados de relleno no sobrepasan el 10 % el total de estados finales de la secuencia a modular y transmitir.

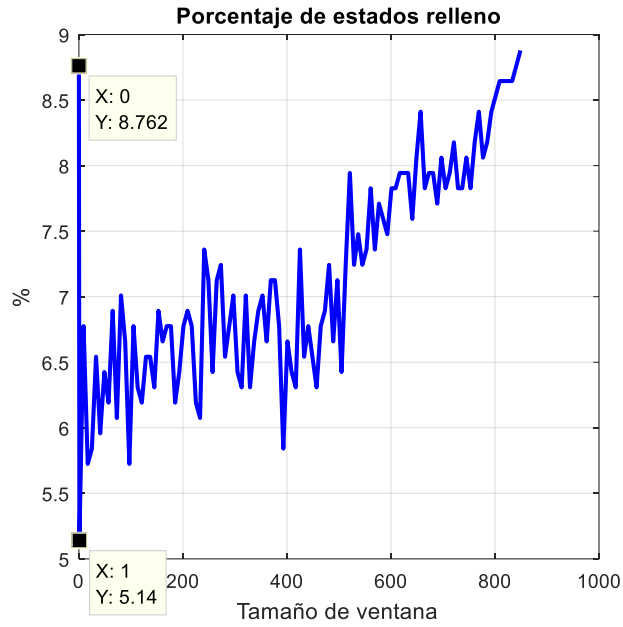


Figura 5.18: Porcentaje de estados de relleno utilizados.

5.8. Campo de acción de BPSK^{+S}

Ahora mostraremos un caso de aplicación de BPSK^{+S} en la comunicación de sensores inalámbricos donde la modulación adaptativa no es muy común y los sensores por lo general presentan potencia de transmisión fija, generalmente se usa la modulación BPSK.

Asumiendo que cierta aplicación funciona con $BER = 10^{-6}$ como mínimo. La Figura 5.20 muestra un esquema donde se pueden ver tres nodos que se comunican. El nodo A puede enviar su información al nodo B en una distancia máxima, donde $BER = 10^{-6}$ modulando con BPSK, sin embargo, cuando el nodo A envía su información al nodo C, el BER disminuye. Esta mejora se debe a que E_b / N_0 (una versión normalizada de la relación señal / ruido) aumenta, pero BPSK no puede aprovechar este evento. Cuando un nodo está a la distancia máxima donde la modulación BPSK^{+S} alcanza un $BER = 10^{-6}$, el nodo puede usar BPSK^{+S} en lugar de BPSK, obteniendo con esto un ahorro energético.

Para mostrar el alcance de nuestra propuesta utilizaremos el modelo de propagación *path loss simplificado* (es mejor utilizar un modelo simple que capte la esencia de la propagación de la señal sin recurrir a complicados modelos de pérdida de trayectoria, que de todos modos son solo aproximaciones del canal real) cuyas fórmulas se resumen en la tabla 5.6, que nos permitirá determinar el radio de cobertura de ambas modulaciones.

Tabla 5.6: Modelo de propagación [80] [81].

Variable	Expresión matemática	Parámetros
Path Loss	$PL = K(dB) + 10 \cdot \gamma \cdot \log_{10} \left[\frac{d}{d_0} \right]$	$d > d_0$ $d_0 = 10 [m]$ γ : exponente de pérdidas del trayecto
Path Loss dependiente de la distancia	$K(dB) = 20 \cdot \log_{10} \left(\frac{4\pi d_0}{\lambda} \right) \quad \lambda = \frac{c}{f}$	$c = 3 \times 10^8 \left[\frac{m}{s} \right]$
Potencia recibida	$P_{Rx}[dBm] = EIRP[dBm] + G_{Rx}[dBd] - PL[dB]$	
Effective Isotropic Radiated Power	$EIRP[dBm] = P_{Tx}[dBm] + G_{Tx}[dBd] - L_c[dB]$	L_c : pérdidas de la línea
Relación señal a ruido (SNR)	$SNR[dB] = P_{Rx}[dBm] - N[dBm]$	
Ruido térmico (N)	$N[dBm] = 10 \cdot \log_{10}(T \cdot BW \cdot K_0) + NF[dB] + 30$	$T = 290 K$ $K_0 = 1.38 \times 10^{-23} [J/K]$ BW : ancho de banda NF : figura de ruido

Como ejemplo de nodo sensor se utilizaron los módulos DIGI Xbee Pro S2C 802.15.4 RF [73], módulos de bajo costo y que según la hoja de datos del fabricante cuentan con una potencia de transmisión para espacios exteriores de 63 mW (18 dBm), una sensibilidad de recepción de -101 dBm, un alcance de 3200 metros y utilizan la antena dipolo de ganancia 2.1 dBi con conector RP-SMA de 4.5 pulgadas modelo A24-HASM-450 a la frecuencia de 2.4 GHz. Ver figura 5.19.

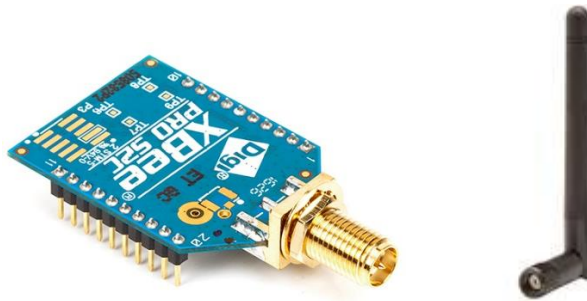


Figura 5.19: Módulo Xbee Pro S2C y antena dipolo A24-HASM-450.

A partir del modelo de propagación, las características del módulo sensor y la SNR mínima teórica para BPSK (10.5 dB) y BPSK^{+S} (16.6 dB) calculadas en el epígrafe 3.4.3, podemos estimar la cobertura como sigue.

Primeramente determinamos el ruido térmico:

$$\begin{aligned}
 N &= 10 \cdot \log_{10}(T \cdot BW \cdot K_0) + NF[dB] + 30 & (5.1) \\
 &= 10 \cdot \log_{10} \left(290[K] \cdot 5 \times 10^6[Hz] \cdot 1.38 \times 10^{-23} \left[\frac{J}{K} \right] \right) + 4 [dB] + 30 \\
 &= -102.9875 [dBm]
 \end{aligned}$$

Luego la potencia de recepción asumiendo $SNR = 10.5 dB$ para BPSK es:

$$\begin{aligned}
 P_{Rx} &= SNR[dB] - N[dBm] & (5.2) \\
 &= 10.5 [dB] + (-102.9875 [dBm]) \\
 &= -92.4875 [dBm]
 \end{aligned}$$

Se obtiene el *Path Loss* dependiente de la distancia (K) tomando como referencia $d_0 = 10$ [m] por ser para exteriores el cálculo (rango de 10 a 100 metros para valor de d_0):

$$\lambda = \frac{c}{f} = \frac{3 \times 10^8 \left[\frac{m}{s} \right]}{2.42 \times 10^9} = 0.124 \text{ [m]} \quad (5.3)$$

$$K = 20 \cdot \log_{10} \left(\frac{4\pi d_0}{\lambda} \right) = 20 \cdot \log_{10} \left(\frac{4\pi(10[m])}{0.124 [m]} \right) = 60.116 \text{ [dB]} \quad (5.4)$$

De la ecuación de potencia de recepción, despejando el *Path Loss* queda:

$$PL = EIRP[dBm] + G_{Rx}[dBd] - P_{Rx}[dBm] \quad (5.5)$$

Sin embargo, aún falta obtener el valor EIRP, por ello:

$$\begin{aligned} EIRP &= P_{Tx}[dBm] + G_{Tx}[dBd] - L_c[dB] \\ &= 18 [dBm] + 4.25 [dBd] - 0.5 dB \\ &= 21.75 [dBm] \end{aligned} \quad (5.6)$$

Sustituyendo ecuación 5.6 en 5.5 queda:

$$PL = EIRP[dBm] + G_{Rx}[dBd] - P_{Rx}[dBm] = 21.75 + 4.25 - (-92.4875) = 118.488 [dBm] \quad (5.7)$$

Con el resultado de ecuación 5.7 es posible despejar el valor de (d) en la ecuación del *Path Loss* tomando como valor de exponente de pérdidas del trayecto $\gamma = 2.7$:

$$PL = K(dB) + 10 \cdot \gamma \cdot \log_{10} \left[\frac{d}{d_0} \right] \quad (5.8)$$

$$118.488 [dBm] = 60.116 [dB] + 10 \cdot 2.7 \cdot \log_{10} \left[\frac{d}{10 [m]} \right]$$

$$d = 1451.77 \text{ [m]}$$

Repitiendo el procedimiento para la SNR de BPSK^{+S} queda:

$$d = 862.98 \text{ [m]}$$

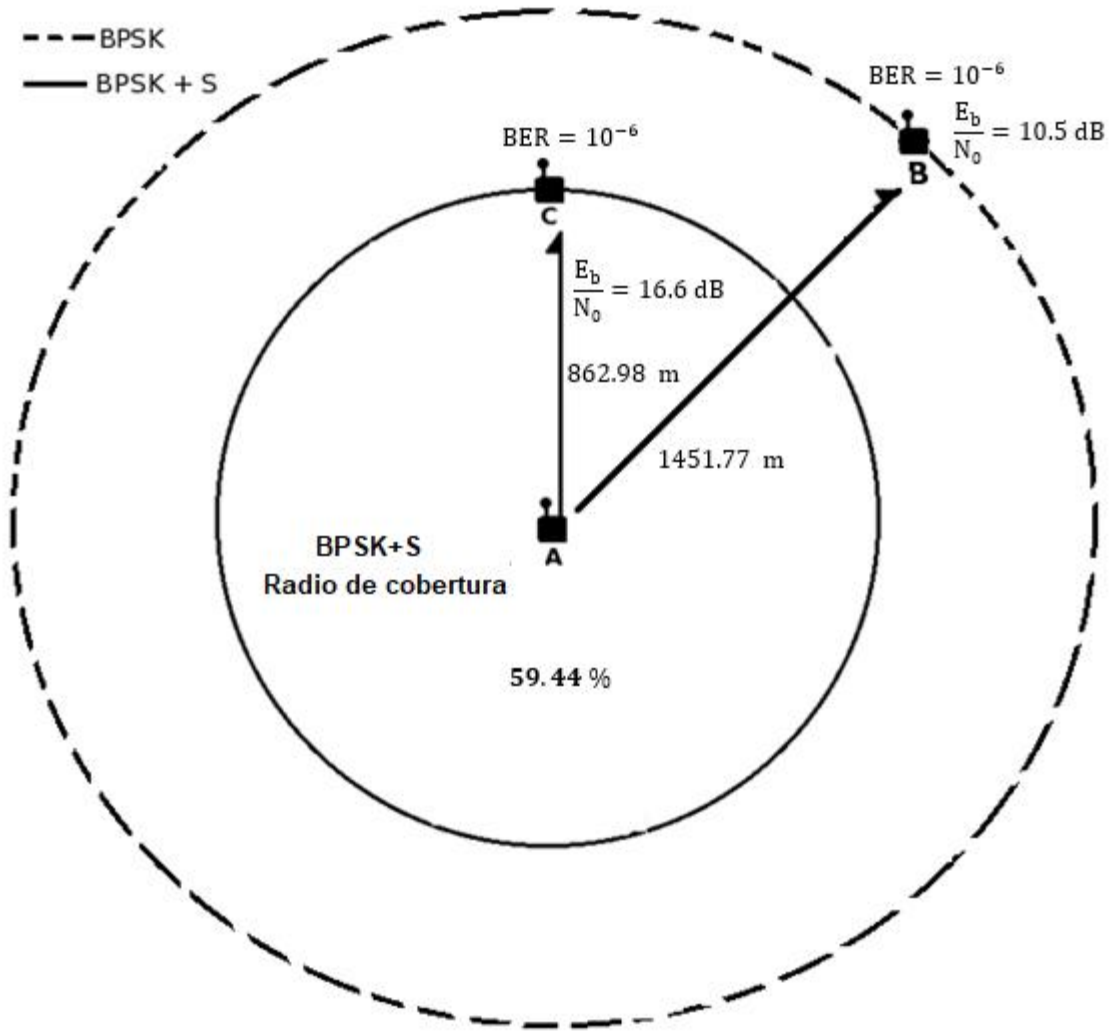


Figura 5.20: Radio de cobertura modulaciones BPSK^{+S} y BPSK.

De la figura 5.20 se aprecia que el alcance de un nodo sensor con modulación BPSK^{+S} se reduce en un 40 % aproximadamente con respecto al empleo de la modulación BPSK en base al radio de transmisión. Sin embargo es importante destacar el ahorro energético que supone utilizar BPSK^{+S} cuando el radio de cobertura lo permite.

Capítulo 6

Conclusiones

En el presente trabajo de tesis se desarrolló e implementó la nueva técnica de codificación-modulación BPSK^{+S} en los radios NI USRP 2932. Se constató la importancia de plataformas como LabVIEW y MATLAB para el trabajo con los mismos, por su facilidad de empleo. Se demostró el gran potencial de los SDR para definir un sistema de comunicaciones inalámbricas, donde las funciones principales de capa física pueden ser implementadas y evaluadas sin ningún cambio de *hardware*.

Según el estado del arte para la presente investigación, muchos esfuerzos se han destinado a tratar de optimizar el consumo energético en los nodos sensores pertenecientes a una red inalámbrica de sensores. Todo ello motivó el desarrollo de una adaptación a la modulación BPSK utilizada en el estándar IEEE 802.15.4, la cual se caracterizó a partir de varios parámetros que permitieron definir su mejor desempeño.

Se advirtió dada la probabilidad de error calculada teóricamente y el BER determinado a partir de los experimentos, que ambos valores se encuentran aproximadamente 6 dB por encima de la SNR necesaria para la modulación BPSK con una tasa de error de bits de 10^{-6} ; 16.6 dB de relación señal a ruido para la modulación BPSK^{+S} (ver figura 3.19 y figura 5.7).

La nueva estrategia de codificación demostró tener puntos a favor como la reducción de un 15 % del total de bits iniciales de la secuencia de datos, siempre que se fije el valor de ventana óptimo ($w=1$). Existe cerca del 50 % de estados S o silencios en la secuencia final modulada para el tamaño óptimo de ventana y en el peor de los casos que sea necesario debido a los factores que afectan el canal de comunicaciones establecer un tamaño de ventana de codificación de valor cero, el 37 % continúa siendo estados S.

Un aspecto negativo de la codificación BPSK^{+S} evidenciado en la implementación fue el llamado error de bit en avalancha o efecto BERA, que ocasionó del lado receptor un aumento del BER y la imposibilidad de una correcta decodificación. En este sentido se determinó el uso de un tamaño adecuado de bloque de codificación ($m=24$) o valores cercanos, valor obtenido a partir de experimentos con múltiples secuencias aleatorias de datos.

La sincronización entre transmisor y receptor es un aspecto vital en las modulaciones digitales, debido a que la transmisión de un estado S o silencio representa la ausencia de señal portadora por el tiempo de duración del símbolo, una secuencia de estados S seguidos afecta la sincronización del sistema BPSK^{+S} y con ello la correcta demodulación. Por resultados experimentales se determinó como valor óptimo de estados S seguidos permitidos tres o menos.

Constituyó una dificultad estimar adecuadamente el ahorro energético representado por el nuevo esquema de codificación-modulación. Las SDR son una plataforma con múltiples ventajas en cuanto a desarrollo e implementación de los esquemas de modulación, pero no cuentan con herramientas de medición de parámetros de consumo energético.

En general, se puede concluir que la técnica de codificación-modulación BPSK^{+S}, es aplicable en nodos sensores dentro de una WSN y dispositivos del internet de las cosas como mecanismo de optimización del

consumo energético solo en condiciones muy específicas, dígame, relación señal a ruido superior a 16.6 dB aproximadamente, así como dispositivos que por su simplicidad no cuenten con mecanismos de adaptación de potencia (presentan una potencia de transmisión fija).

A.1. Código función de probabilidad de error de bit (MATLAB)

```
% modulations transmission over AWGN channel
close all;
clear all;
clc;
SNRdBbpsk=0:0.1:10.5;           %SNR in dB
SNRdBbook=0:0.1:13.5;         %SNR in dB
SNRdBbpsks=0:0.1:16.6;        %SNR in dB
SNRbpsk=10.^(SNRdBbpsk/10);    %SNR in linear scale
SNRbook=10.^(SNRdBbook/10);    %SNR in linear scale
SNRbpsks=10.^(SNRdBbpsks/10);  %SNR in linear scale

%BPSK
semilogy(SNRdBbpsk,'k-', 'Linewidth',2.0);
hold on
%OOK
semilogy(SNRdBbook,'k--', 'Linewidth',2.0);
hold on
%BPSK+S
semilogy(SNRdBbpsks, 'k-', 'Linewidth',2.0);
title('Probability of Bit Error over AWGN')
xlabel('Eb/No in dB')
ylabel('BER')
legend('BPSK', 'OOK', 'BPSK+S')
grid on
```

A.2. Código función codificador BPSK^{+S} (MATLAB)

```

clear C;
clear new_pakete;
clear paketeeee;
clear pakete_datos;
clear paketeunido;
clear pakete_finalcode;
clear paketeaux;

%PARAMETROS CONFIGURABLES
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tam_seg      %tamaño de los bloques de segmento en bits
%xsilen      %cada cuan"o" silencios pongo el bit "1"
%ven         %tamaño de ventana en bits

%BITS DE GUARDA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
M_unos=ones(1,8);           % los 8 primeros bits en unos
M_ceros=zeros(1,4);        % los 4 siguientes bits en ceros

%TAMAÑO DE LA VENTANA EN BITS (min 2 máximo 1024)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tam_ventana_decimal=ven;   %valor del salto
binario=de2bi(tam_ventana_decimal) %convierto a binario el tam ventana
tam=size(binario);
M_tam_ventana=[binario,zeros(1,16-tam(2))]; %relleno con ceros para q sea un campo de 8
bits

%TAMAÑO DEL PAKETE EN BYTES (min 2 máximo 255)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tt=size(aux);
tam_pakete_decimal=(tt(2))/8; %+8 bits de relleno al final por el radio
binario2=de2bi(tam_pakete_decimal); %82unter82rto a binario el tam pakete
tam2=size(binario2);
M_tam_pakete=[binario2,zeros(1,8-tam2(2))]; %%%%%%%%% %rellenos con ceros para q sea un
campo de 8 bits

%BITS DE INICIO DE CABECERA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
start=[0 1 1 1 1 1 0 0 1 1 1 1 1 1"0]" % se toma el caracter "~" para delimitar inicio
de cabecera

%MATRIZ FINAL PARA EVITAR SE PIERDAN DATOS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
M_tresceros=[0 0 0 1 1 1 1 1];

%SECUENCIA PSEUDOALEATORIA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
secuencia_aleatoria=[1,1,1,1,0,0,1,1,0,1,1,0,1,0,1,0,0,1,0,0]; %20 bits

%PAKETE FINAL ANTES DE CODIFICAR CON CABECERA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
cabecera=[M_unos,M_ceros,secuencia_aleatoria,start,M_tam_ventan82unter82m_pakete];
%cabecera
pakete_final=[cabecera,aux,M_tr82unter82s]; %paquete final
pakete_datos=aux; %paquete de datos

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%CODIFICACION CON SILENCIOS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%-%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 1 -- tx valor de tam_ntana hacia adelante
% 0 -- tx un silencio en83unter83caso un 5

paketeeee=pakete_datos;          %solo codifico con silencios el paquete de datos
tam=size(paketeeee);
tam_pakete_bit=tam(2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%SEGMENTADO DEL PAKETE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
exp=tam_pakete_bit/tam_seg;
if mod(tam_pakete_bit,tam_seg)==0          %determ83unteranto83unter838383iónn se duvide
la codificacion
    partes=floor(exp);
else
    partes=floor(exp)+1;
end
paketeunido=[];
bitsresto=tam_pakete_bit;          %inicio en tamaño del paquete
bitsrec=0;
for seg=1:partes
    %analizar como irme moviendo por los pedazos de paquete
    if (bitsresto)>=tam_seg          %determino tamaño del segmento por si es el final q puede
ser mas corto
        tam_pakete_bits=tam_seg;
        paquete=paketeeee(bitsrec+1:bitsrec+tam_seg);
    else
        tam_pakete_bits=bitsresto;
        paquete=paketeeee(bitsrec+1:bitsrec+bitsresto);
    end
    bitsrec=(seg)*tam_seg;          %cantidad de bits acumulados
    bitsresto=tam_pakete_bit-bitsrec; %bits que van quedando
    cont1=1; % contador para el vector posiciones
    cont1=1; % contador para el vector new_pakete
    cont2=1; % contador para comparar con las posiciones
    new_pakete=0;
    posicion= zeros(1,tam_pakete_bit);
    for i=1:tam_pakete_bits
        if paquete(i)== 0 && posicion(cont2)~= i
            new_pakete(cont1)=5;
            cont1=cont1+1;
        else
            if posicion(cont2)~= i
                if i+tam_ventana_decimal <= tam_pakete_bits
                    new_pakete(cont1)=paquete(i+tam_ventana_dec83unter8383
posicion(cont)=i+tam_ventana_decimal;
                    cont=cont+1;
                else
                    new_pakete(cont1)=paquete(i);
                end
                cont1=cont1+1;
            else
                cont2=cont2+1;
            end
        end
    end
    paqueteunido=[paketeunido,new_pakete]; %uno los segmentos codificados en uno solo
end
new_pakete=paketeunido;

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%AGREGAR UN "1" LUEGO DE X SILENCIOS SEGUIDOS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tamnewpak=size(new_pakete);
preC=tamnewpak(2);
contsilen=0;
ee=1;
cantbitsbasura=0;
cantidad_silencios=0;
for e=1:preC
    paketeaux(ee)=new_pakete(e);
    if new_pakete(e)==5
        contsilen=contsilen+1; %cuento la cantidad silencios seguidos
        if contsilen==xsilen
            ee=ee+1;
            paketeaux(ee)=1;
            cantbitsbasura=cantbitsbasura+1;
            contsilen=0;
        end

        cantidad_silencios=cantidad_silencios+1;
    else
        contsilen=0;
    end
    ee=ee+1;
end
pakete_finalcode=[cabecera,paketeaux,M_tresceros];

%SALIDA DE DATOS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C=pakete_finalcode;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.3. Código función decodificador BPSK^{+S} (MATLAB)

```

%PARAMETROS CONFIGURABLES
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tam_seg          %tamaño del segmento en bits
%xsilen           %cada cuantos silencios incorporo un 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%decodificación con silencios
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%-%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 5 -- silencio-equivalente a un cero
% 0 -- bit en (1) en esa posición y en (0) x p-iciones más adelante
% 1 -- bit en (1) en esa posición y en (1) x posiciones más adelante

%MECANISMO DE DETECCION DE LA CABECERA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%aquí determino a partir de donde empieza la cabecera
sec_inicio=[0 1 1 1 1 1 0 0 1 1 1 1 0];
tama=size(sec_inicio);
tamaa=size(aa);
cc=0;
ini=1;
control=0;
bitrec=0;
while 1
    for x=1:tama(2)
        if sec_inicio(x)==aa(ini)
            ini=ini+1;
            control=control+1;
            bitrec=bitrec+1;
        else
            ini=ini-bitrec+1;
            control=0;
            bitrec=0;
            break;
        end
    end
    if control==tama(2)
        break;
    end
    if (ini+24)>tamaa(2)
        break
    end
end
if control==tama(2) && (ini+24)<=tamaa(2)
    w=true;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for g=1:24
        cabecera_rec(g)=aa(ini);85unte      %determino si hay algun 5 en la cabecera
        if cabecera_rec(g)==5
            w=false;
            break;
        end
        ini=ini+1;
    end

    if w
        inil=ini; %guardo inicio del paquete de datos
%DETERMINAR EL TAMAÑO DE LA VENTANA Y DEL PAQUETE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        tamwinbin=cabecera_rec(1:16);
        tampakbin=cabecera_rec(17:24);

```



```

%%%%%%%%%%
cc=bi2de(tampakbin);
tam_pakete_bits=bi2de(tampakbin)*8;%para convertirlo a decimal
tam_ventana_decimal=bi2de(tamwinbin); %
aa=aa(ini1:end); %datos codificados
%determino el tamaño de los datos codificados
tamm=size(aa)
tam_new_pakete=tamm(2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%ELIMINAR EL BIT BASURA DESPUES DE X SILENCIOS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
contsilen=0; %%%%%%%%%%
p=1;
for j=1:tam_new_pakete
    eva=aa(j);
    if eva==5
        if contsilen==xsilenn
            contsilen=0;
        else
            out(p)=eva;
            contsilen=contsilen+1;
            p=p+1;
        end
    elseif eva==0
        if contsilen==xsilenn
            contsilen=0;
        else
            out(p)=eva;
            contsilen=0;
            p=p+1;
        end
    else
        if contsilen==xsilenn
            contsilen=0;
        else
            out(p)=eva;
            contsilen=0;
            p=p+1;
        end
    end
end
end
aa=out;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%SEGMENTADO DEL PAQUETE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
exp=tam_pakete_bits/tam_bloque;
if mod(tam_pakete_bits,tam_bloque)==0 %determino cuantos segmetos son
    partes=floor(exp);
else
    partes=floor(exp)+1;
end
D=[];
bitsresto=tam_pakete_bits; %inicio en tamaño del paquete
bitsrec=0;
puntero=1; %indica en que parte del paquete me quede
for h=1:partes
    if (bitsresto)>=tam_bloque
        tam_pakete_bit=tam_bloque;
    else
        tam_pakete_bit=bitsresto;
    end
    bitsrec=(h)*tam_bloque;
    bitsresto=tam_pakete_bit-86unter86rec;
end

```

```

pakete_reconstruido=zeros(1,tam_pakete_bit);
conta=1;
contal=1;
comp=0;
pos=0;
a=1;
position= zeros(1,tam_pake87unter;
for j=puntero:tam_new_pakete
    aux=aa(j);
    if aux==5
        while position(conta)==(a+comp)
            comp=comp+1;
            conta=conta+1;
        end
        paquete_reconstruido(a+comp)=0;
        a=a+comp+1;
        comp=0;
    elseif aux==0
        while position(conta)==(a+comp)
            comp=comp+1;
            conta=conta+1;
87unter87
        paquete_reconstruido(a+comp)=1;
        pos=a+comp+tam_ventana_decimal;
        a=a+comp+1;
        if pos <= tam_pakete_b87unter87
pakete_reconstruido(pos)=0;
            position(contal)=pos;
            contal=contal+1;
        end
        comp=0;
    else
        while position(conta)==(a+comp)
            comp=comp+1;
            conta=conta+1;
87unter87
        paquete_reconstruido(a+comp)=1;
        pos=a+comp+tam_ventana_decimal;
        a=a+comp+1;
        if pos <= tam_pakete_b87unter87
pakete_reconstruido(pos)=1;
            position(contal)=pos;
            contal=contal+1;
        end
        comp=0;
    end
    if tam_pakete_bit <=(a-1)
        break;
    end
    end
    taman=size(pakete_reconstruido);
    if taman(2)>tam_87unterbit
        puntero=j;
87unterlse
        puntero=j+1;
    end
    D=[D,pakete_reconstruido(1:tam_pakete_bit)];
end
    %fin del for de cada pedazo
end
else
    D=[0 1 0 0 0 1 0 1 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 1 1 1 0 1 0 1 0 1 0... 0 1 0 0 0
0 0 1 0 1 0];
end

```

A.4. Especificaciones de módulos RF Xbee/Xbee-PRO ZB

Specification	XBee	XBee-PRO (S2)	XBee-PRO (S2B)
Performance			
Indoor/urban range	up to 133 ft. (40 m)	Up to 300 ft. (90 m), up to 200 ft (60 m) international variant	Up to 300 ft. (90 m), up to 200 ft (60 m) international variant
Outdoor RF line-of-sight range	up to 400 ft. (120 m)	Up to 2 miles (3200 m), up to 5000 ft (1500 m) international variant	Up to 2 miles (3200 m), up to 5000 ft (1500 m) international variant
Transmit power output	2 mW (+3dBm), boost mode enabled 1.25 mW (+1dBm), boost mode disabled	50 mW (+17 dBm) 10 mW (+10 dBm) for International variant	63mW (+18 dBm) 10mW (+10 dBm) for International variant
RF data rate	250,000 b/s	250,000 b/s	250,000 b/s
Data throughput	up to 35000 b/s (see Transmission, addressing, and routing)	up to 35000 b/s (see Transmission, addressing, and routing)	up to 35000 b/s (see Transmission, addressing, and routing)
Serial interface data rate (software selectable)	1200 b/s - 1 Mb/s (non-standard baud rates also supported)	1200 b/s - 1 Mb/s (non-standard baud rates also supported)	1200 b/s - 1 Mb/s (non-standard baud rates also supported)
Receiver sensitivity	-96 dBm, boost mode enabled -95 dBm, boost mode disabled	-102 dBm	-102 dBm

Power Requirements			
Supply voltage	2.1 - 3.6 V	3.0 - 3.4 V	2.7 - 3.6 V
Operating current (transmit, max output power)	40 mA (@ 3.3 V, boost mode enabled) 35 mA (@ 3.3 V, boost mode disabled)	295 mA (@3.3 V) 170 mA (@3.3 V) international variant	205 mA, up to 220 mA with programmable variant (@3.3 V) 117 mA, up to 132 mA with programmable variant (@3.3 V), International variant
Operating current (receive)	40mA (@ 3.3 V, boost mode enabled) 38mA (@ 3.3 V, boost mode disabled)	45 mA (@3.3 V)	47 mA, up to 62 mA with programmable variant (@3.3 V)
Idle current (receiver off)	15mA	15mA	15mA
Power-down current	< 1 uA @ 25°C	3.5 uA typical @ 25°C	3.5 uA typical @ 25°C

General			
Operating frequency band	ISM 2.4 GHz	ISM 2.4 GHz	ISM 2.4 GHz
Dimensions	0.960" x 1.087" (2.438cm x 2.761cm)	0.960 x 1.297 (2.438 cm x 3.294 cm)	0.960 x 1.297 (2.438 cm x 3.294 cm)
Operating temperature	-40 to 85° C (industrial)	-40 to 85° C (industrial)	-40 to 85° C (industrial)
Antenna options	Integrated whip antenna, embedded PCB antenna, RPSMA, or U.FL connector	Integrated whip antenna, embedded PCB antenna, RPSMA or U.FL connector	Integrated whip antenna, embedded PCB antenna, RPSMA or U.FL connector
I/O interface	3.3 V CMOS UART (not 5 V tolerant), DIO, ADC	3.3 V CMOS UART (not 5 V tolerant), DIO, ADC	3.3 V CMOS UART (not 5 V tolerant), DIO, ADC

A.5. La función error

La función error $erf(x)$ y la función error complementaria $erfc(x)$ se definen mediante las integrales [66]:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-z^2) dz \quad \text{y} \quad erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-z^2) dz$$

Propiedades:

1. $erf(x) + erfc(x) = 1$
2. $erf(\infty) = 1$
3. $erf(-\infty) = -1$
4. $erf(-x) = -erf(x)$
5. $erfc(-\infty) = 2$
6. $erfc(\infty) = 0$
7. $erfc(0) = 1$

En la figura a.1 se muestra $erf(x)$ y $erfc(x)$ desde $-\infty < x < \infty$.

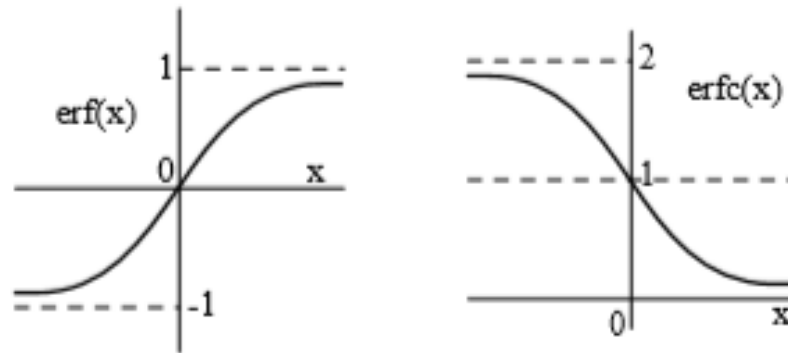


Figura a.1: Función $erf(x)$ y $erfc(x)$.

En algunos textos, en vez de la función $erfc(x)$, se utiliza la función:

$$Q(y) = \frac{1}{\sqrt{2\pi}} \int_y^\infty \exp\left(-\frac{z^2}{2}\right) dz$$

Las funciones $Q(y)$ y $erfc(x)$ se pueden relacionar mediante las expresiones:

$$erfc(x) = 2Q(\sqrt{2x}) \quad \text{o} \quad Q(y) = \frac{1}{2} erfc\left(\frac{y}{\sqrt{2}}\right)$$

En la tabla a.1 se dan los valores de la función $erfc(x)$ para $0 \leq x \leq 3.49$

Tabla a.1: Tabla de valores de $erfc(x)$.

x	<i>erfc(x)</i>	x	<i>erfc(x)</i>	x	<i>erfc(x)</i>	x	<i>erfc(x)</i>	x	<i>erfc(x)</i>	x	<i>erfc(x)</i>	x	<i>erfc(x)</i>
0	1.000000	0.5	0.479500	1	0.157299	1.5	0.033895	2	0.004678	2.5	0.000407	3	0.00002209
0.01	0.988717	0.51	0.470756	1.01	0.153190	1.51	0.032723	2.01	0.004475	2.51	0.000386	3.01	0.00002074
0.03	0.966159	0.53	0.453536	1.03	0.145216	1.53	0.030484	2.03	0.004094	2.53	0.000346	3.03	0.00001827
0.04	0.954889	0.54	0.445061	1.04	0.141350	1.54	0.029414	2.04	0.003914	2.54	0.000328	3.04	0.00001714
0.05	0.943628	0.55	0.436677	1.05	0.137564	1.55	0.028377	2.05	0.003742	2.55	0.000311	3.05	0.00001608
0.06	0.932378	0.56	0.428384	1.06	0.133856	1.56	0.027372	2.06	0.003577	2.56	0.000294	3.06	0.00001508
0.07	0.921142	0.57	0.420184	1.07	0.130227	1.57	0.026397	2.07	0.003418	2.57	0.000278	3.07	0.00001414
0.08	0.909922	0.58	0.412077	1.08	0.126674	1.58	0.025453	2.08	0.003266	2.58	0.000264	3.08	0.00001326
0.09	0.898719	0.59	0.404064	1.09	0.123197	1.59	0.024538	2.09	0.003120	2.59	0.000249	3.09	0.00001243
0.1	0.887537	0.6	0.396144	1.1	0.119795	1.6	0.023652	2.1	0.002979	2.6	0.000236	3.1	0.00001165
0.11	0.876377	0.61	0.388319	1.11	0.116467	1.61	0.022793	2.11	0.002845	2.61	0.000223	3.11	0.00001092
0.12	0.865242	0.62	0.380589	1.12	0.113212	1.62	0.021962	2.12	0.002716	2.62	0.000211	3.12	0.00001023
0.13	0.854133	0.63	0.372954	1.13	0.110029	1.63	0.021157	2.13	0.002593	2.63	0.000200	3.13	0.00000958
0.14	0.843053	0.64	0.365414	1.14	0.106918	1.64	0.020378	2.14	0.002475	2.64	0.000189	3.14	0.00000897
0.15	0.832004	0.65	0.357971	1.15	0.103876	1.65	0.019624	2.15	0.002361	2.65	0.000178	3.15	0.00000840
0.16	0.820988	0.66	0.350623	1.16	0.100904	1.66	0.018895	2.16	0.002253	2.66	0.000169	3.16	0.00000786
0.17	0.810008	0.67	0.343372	1.17	0.098000	1.67	0.018190	2.17	0.002149	2.67	0.000159	3.17	0.00000736
0.18	0.799064	0.68	0.336218	1.18	0.095163	1.68	0.017507	2.18	0.002049	2.68	0.000151	3.18	0.00000689
0.19	0.788160	0.69	0.329160	1.19	0.092392	1.69	0.016847	2.19	0.001954	2.69	0.000142	3.19	0.00000644
0.2	0.777297	0.7	0.322199	1.2	0.089686	1.7	0.016210	2.2	0.001863	2.7	0.000134	3.2	0.00000603
0.21	0.766478	0.71	0.315335	1.21	0.087045	1.71	0.015593	2.21	0.001776	2.71	0.000127	3.21	0.00000564
0.23	0.744977	0.73	0.301896	1.23	0.081950	1.73	0.014422	2.23	0.001612	2.73	0.000113	3.23	0.00000493
0.24	0.734300	0.74	0.295322	1.24	0.079495	1.74	0.013865	2.24	0.001536	2.74	0.000107	3.24	0.00000460
0.25	0.723674	0.75	0.288845	1.25	0.077100	1.75	0.013328	2.25	0.001463	2.75	0.000101	3.25	0.00000430
0.26	0.713100	0.76	0.282463	1.26	0.074764	1.76	0.012810	2.26	0.001393	2.76	0.000095	3.26	0.00000402
0.27	0.702582	0.77	0.276179	1.27	0.072486	1.77	0.012309	2.27	0.001326	2.77	0.000090	3.27	0.00000376
0.28	0.692120	0.78	0.269990	1.28	0.070266	1.78	0.011826	2.28	0.001262	2.78	0.000084	3.28	0.00000351
0.29	0.681717	0.79	0.263897	1.29	0.068101	1.79	0.011359	2.29	0.001201	2.79	0.000080	3.29	0.00000328
0.3	0.671373	0.8	0.257899	1.3	0.065992	1.8	0.010909	2.3	0.001143	2.8	0.000075	3.3	0.00000306
0.31	0.661092	0.81	0.251997	1.31	0.063937	1.81	0.010475	2.31	0.001088	2.81	0.000071	3.31	0.00000285
0.33	0.640721	0.83	0.240476	1.33	0.059985	1.83	0.009653	2.33	0.000984	2.83	0.000063	3.33	0.00000249
0.34	0.630635	0.84	0.234857	1.34	0.058086	1.84	0.009264	2.34	0.000935	2.84	0.000059	3.34	0.00000232
0.35	0.620618	0.85	0.229332	1.35	0.056238	1.85	0.008889	2.35	0.000889	2.85	0.000056	3.35	0.00000216
0.36	0.610670	0.86	0.223900	1.36	0.054439	1.86	0.008528	2.36	0.000845	2.86	0.000052	3.36	0.00000202
0.37	0.600794	0.87	0.218560	1.37	0.052688	1.87	0.008179	2.37	0.000803	2.87	0.000049	3.37	0.00000188
0.38	0.590991	0.88	0.213313	1.38	0.050984	1.88	0.007844	2.38	0.000763	2.88	0.000046	3.38	0.00000175
0.39	0.581261	0.89	0.208157	1.39	0.049327	1.89	0.007521	2.39	0.000725	2.89	0.000044	3.39	0.00000163
0.4	0.571608	0.9	0.203092	1.4	0.047715	1.9	0.007210	2.4	0.000689	2.9	0.000041	3.4	0.00000152
0.41	0.562031	0.91	0.198117	1.41	0.046148	1.91	0.006910	2.41	0.000654	2.91	0.000039	3.41	0.00000142
0.43	0.543113	0.93	0.188437	1.43	0.043143	1.93	0.006344	2.43	0.000589	2.93	0.000034	3.43	0.00000123
0.44	0.533775	0.94	0.183729	1.44	0.041703	1.94	0.006077	2.44	0.000559	2.94	0.000032	3.44	0.00000115
0.45	0.524518	0.95	0.179109	1.45	0.040305	1.95	0.005821	2.45	0.000531	2.95	0.000030	3.45	0.00000107
0.46	0.515345	0.96	0.174576	1.46	0.038946	1.96	0.005574	2.46	0.000503	2.96	0.000028	3.46	0.00000099
0.47	0.506255	0.97	0.170130	1.47	0.037627	1.97	0.005336	2.47	0.000477	2.97	0.000027	3.47	0.00000092
0.48	0.497250	0.98	0.165769	1.48	0.036346	1.98	0.005108	2.48	0.000453	2.98	0.000025	3.48	0.00000086
0.49	0.488332	0.99	0.161492	1.49	0.035102	1.99	0.004889	2.49	0.000429	2.99	0.000024	3.49	0.00000080

A.6. Diagrama de maqueta



NI USRP 2932

Parámetros	Transmisor	Receptor
Ganancia de transmisión	0-31 dB	
Ganancia de recepción		0 dB
Dirección IP	192.168.10.5	192.168.11.2
Muestras por símbolo	8	8
Modulación	BPSK ^{+S}	BPSK ^{+S}
Distancia	3 metros	

PC

Parámetros	
Sistema Operativo	Windows 10 64 bits
Microprocesador	Intel core™ i5 CPU
Memoria RAM	8 Gb
Conexión de red	Gigabit Ethernet
Software aplicación	LabVIEW 2017
Software NI USRP	NI USRP 29XX

- [1] A. Moschitta and IgNeri. (2014) «Power consumption Assessment in Wireless Sensor Networks». Intech 2014.
- [2] A. Villalba Macías (2015) « Aplicación de una red inalámbrica de sensores para detección de presencia y conteo», Sevilla, España, pp. 9.
- [3] Bustillo, O. J. (2016). « Estudio de técnicas de Modulación mediante Radios NI USRP». Trabajo de grado, Ingeniería en telecomunicaciones, Universidad Nacional Autónoma de México, México.
- [4] P. Rosero, S. Núñez, S. Realpe, V. Alvear, L. Beltrán y C. Rosado. (2017) «Internet de las cosas y redes de sensores inalámbricos: review», pp. 1-7.
- [5] J. Carlos Reyes Guerrero. (2012) « Localización en Interiores Usando Redes Inalámbricas de Sensores », ISBN 3848463687.
- [6] W. K. Seah, Z. A. Eu & H. P. Tan, (2009). «Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges». In *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*, 2009. Wireless VITAE 2009, pp. 1-5. IEEE.
- [7] J. R. Lajara Vizcaíno. (Junio 2014), «Modelado y optimización de energía en redes de sensores inalámbricas para la medida de parámetros medioambientales», Tesis doctoral Universidad Politécnica de Valencia. Pp. 25-42.
- [8] Ye, W., Heidemann, J., & Estrin, D. (2002). «An energy-efficient MAC protocol for wireless sensor networks». In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings.* IEEE ,Vol. 3, pp. 1567-1576. IEEE.
- [9] Polastre, J., Hill, J., & Culler, D. (2004). «Versatile low power media access for wireless sensor networks». In *Proceedings of the 2nd international conference on Embedded networked sensor systems* (pp. 95-107). ACM.
- [10] El-Hoiydi, A., Decotignie, J. D., Enz, C., & Le Roux, E. (2003). «Wisemac, an ultra low power mac protocol for the wisenet wireless sensor network». In *Proceedings of the 1st international conference on Embedded networked sensor systems* (pp. 302-303). ACM.
- [11] Cheng Yin, Ya Li & Dongquan Zhang. (2010) « DSMAC: An energy-efficient MAC protocol in Event-driven Sensor Networks». In 0, pp. 422-425. IEEE.

- [12] I. Lee & K. Lee. (2015). «TheInternetofThings(IoT):Applications, investments, and challenges for enterprises», *Business Horizons*, vol. 58, pp. 431— 440.
- [13] H. B. Pandya & T. A. Champaneria. (2015) «Internet of things: Survey and case studies», de *International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO)*, Visakhapatnam.
- [14] X. Chen, X. Wang & X. Chen. (2013) «Energy-Efficient Optimization for Wireless Information and Power Transfer in Large-Scale MIMO Systems Employing Energy Beamforming», *IEEE Wireless Communications Letters*, vol. 2,nº 6, pp. 667 – 670.
- [15] M. Magno , D. Boyle, D. Brunelli & E. Popovici. (2014) «Ensuring Survivability of Resource-Intensive Sensor Networks Through Ultra-Low Power Overlays», *IEEE Transactions on Industrial Informatics*, vol. 10, nº 2, pp. 946 – 956.
- [16] LoRa Alliance. (2015) «A technical overview of LoRa® and LoRaWAN™,» *Technical Marketing Workgroup 1.0*.
- [17] ZigBee Alliance, «The ZigBee Alliance,» [Online]. Available: <http://www.zigbee.org/>.
- [18] IEEE. (2013) «IEEE 802.15.4 Part 15.4: Wireless Medium Access Part 15.4: Wireless Medium Access Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs),» *IEEE Computer Society*, New York, USA.
- [19] H. Karl & A. Willig, (2005). «Protocols and Architectures for Wireless Sensor Networks». In John Wiley & Sons, ltd. (ed.).
- [20] International Telecommunication Union, (2010) «Serie Y.2121: Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment», *Telecommunication Standardization Sector of ITU*.
- [21] International Telecommunication Union, (2010) «Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment (Y.2221) », *Telecommunication Standardization Sector of ITU*.
- [22] I. Akyildiz, W. Su, Y. Sankarasubramaniam y E. Cayirci, (2002) «Wireless sensor networks: a survey», *Computer Networks*, vol. 38, pp. 393-422.
- [23] G.Werner-Allen , K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees & M. Welsh, (2006). «Deploying a wireless sensor network on an active volcano». *Internet Computing, IEEE*, 10(2), pp. 18-25.
- [24] A. Matese, S. F. Di Gennaro, A. Zaldei, L. Genesio & F. P. Vaccari, (2009). «A wireless sensor network for precision viticulture: The NAV system». *Computers and Electronics in Agriculture*, pp. 51-58.
- [25] L. Yu, N. Wang & X. Meng, (2005). «Real-time forest fire detection with wireless sensor networks». In *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on (Vol. 2, pp. 1214-1217)*. IEEE.

- [26] B. Son, Y. S. Her & J. Kim, (2006). «A design and implementation of forest-fires surveillance system based on wireless sensor networks for South Korea mountains». *International Journal of Computer Science and Network Security (IJCSNS)*, pp. 124-130.
- [27] McCulloch, J., McCarthy, P., Guru, S. M., Peng, W., Hugo, D., & Terhorst, A. (2008). «Wireless sensor network deployment for water use efficiency in irrigation». In *Proceedings of the workshop on Real-world wireless sensor networks* (pp. 46-50). ACM.
- [28] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, Gordon & S. Jha, (2006). « Wireless sensor networks for battlefield surveillance ». In *Proceedings of the land warfare conference*.
- [29] S. Puri, (2012). « Intelligent Wireless Sensor Network System to shrink Suspected Terror from Militants». In *International Journal of Computer Network & Information Security*.
- [30] R. Lin, Z. Wang, & Y. Sun, (2004, June). « Wireless sensor networks solutions for real time monitoring of nuclear power plant ». In *Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on* (Vol. 4, pp. 3663-3667). IEEE.
- [31] Hue, personal wireless lighting. Available online: <http://www.meethue.com/es-ES/>
- [32] Wheeler, A. (2007). « Commercial applications of wireless sensor networks using ZigBee ». *Communications Magazine, IEEE*, pp. 70-77.
- [33] C. R. Baker, K. Armijo, S. Belka, M. Benhabib & P. K Wright. (2007). « Wireless sensor networks for home health care ». In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on* (Vol. 2, pp. 832-837). IEEE.
- [34] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu & J. Stankovic, (2006). «ALARM-NET: Wireless sensor networks for assisted-living and residential monitoring ». *University of Virginia Computer Science Department Technical Report*.
- [35] K. S. Low, W. N. N. Win, & M. J. Er, (2005, November). *Wireless sensor networks for industrial environments*. In *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on* (Vol. 2, pp. 271-276). IEEE
- [36] Car 2 Car Communication Consortium. (2007). Braunschweig, November. Available online: <http://www.car-to-car.org/>
- [37] Institute of Electrical and Electronic Engineers. Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY). Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs).
- [38] IEEE. IEEE 802.15.4a-2007 (2007) Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs).
- [39] Institute of Electrical and Electronic Engineers. 802.15.4c-2009 – IEEE Standard for Local and Metropolitan Area Networks – Specific Requirements – Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks

- (WPANs). Amendment 2: Alternative Physical Layer Extension to support one or more of the Chinese 314–316 MHz, 430–434 MHz, and 779–787 MHz.
- [40] Institute of Electrical and Electronic Engineers. 802.15.4d-2009 – IEEE Standard for Local and Metropolitan Area Networks – Specific Requirements – Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs). Amendment 3: Alternative Physical Layer Extension to support the Japanese 950 MHz bands.
- [41] Kazem Sohraby, Daniel Minoli, & Taieb Znati. (2007) *Wireless Sensor Networks: technology, protocols, and applications*. Wiley, 2007 edition. Pp. 176-179.
- [42] Jianping Song, Song Han, Aloysius K Mok, Deji Chen, Mike Lucas, Mark Nixon, & Wally Pratt. (2008) *WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control*. Real-Time and Embedded Technology and Applications Symposium, pages 377–386.
- [43] N. Kushalnagar, G. Montenegro & C. Schumacher. RFC 4919. Ipv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. Available online: <https://datatracker.ietf.org/doc/rfc4919/>
- [44] Data sheet modulo JN5139-yyy-Myy. Available online: <https://media.digikey.com/pdf/.../JN5139-yyy-Myy.pdf>
- [45] *Digi* Technology, Hoja de datos modulos Zigbee. Available online: <https://www.digi.com/resources/documentation/digidocs/PDFs/90000976.pdf>
- [46] L. Iacono, P. Godoy, O. Marianetti y C. García, (2010). « Estudio de Plataformas de Hardware Empleadas en Redes de Sensores Inalámbricas ». XVI Congreso Argentino de Ciencias de la Computación.
- [47] J. Rodríguez, (2015). « Sistema de comunicación inalámbrico basado en el protocolo Zigbee ». Tesis de grado sistemas de comunicaciones. Universidad Carlos III Madrid. Pp. 21-28.
- [48] H. Karvonen & C. Pomalaza-Raez, (2006). « A cross layer design of coding and awake/sleep periods in WSNs ». In 2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 15.
- [49] W. K. Seah, Z. A. Eu & H. P. Tan, (2009). « Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges ». In *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on* (pp. 1-5). IEEE.
- [50] G. W. Taylor, J. R. Burns, S. A. Kammann, W. B. Powers & T. R. Welsh, (2001). « The energy harvesting Eel: a small subsurface ocean/river power generator ». *Oceanic Engineering, IEEE Journal of*, 26(4), pp. 539-547.
- [51] T. Sogorb, J. Vicente Llario, J. Pelegri, R. Lajara & J. Alberola, (2008). « Studying the feasibility of energy harvesting from broadcast RF station for WSN ». In *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE*. pp. 1360-1363.

- [52] B. Ivey, (2011). « Low-Power Design Guide. Microchip application note 1416». Available online: <http://ww1.microchip.com/downloads/en/AppNotes/01416a.pdf>
- [53] Microchip Technology. Compiled Tips ‘N Tricks Guide. Available online: <http://ww1.microchip.com/downloads/en/DeviceDoc/01146B.pdf>
- [54] L. Zhao, W.Zhang, , C. Xu, Y. Xu & X. Li, (2006). « Energy-aware system design for wireless sensor network ». ACTA Automatica Sinica, 32(6), 892.
- [55] V. Raghunathan, C. Schurgers, S. Park & M. B. Srivastava, (2002). « Energy-aware wireless microsensor networks ». Signal Processing Magazine, IEEE,19(2), pp. 40-50.
- [56] G. J. Pottie & W. J. Kaiser, (2000). « Wireless integrated network sensors ». Communications of the ACM, 43(5), pp. 51-58.
- [57] B. V. Reddy & D. R. K. Reddy, (2013). « Maximizing the usable battery capacity in WSN with load harmonization ». International Journal of Instrumentation Electrical Electronics Engineering.
- [58] W. Ye, J. Heidemann & D. Estrin, (2002). « An energy-efficient MAC protocol for wireless sensor networks ». In INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE (Vol. 3, pp. 1567-1576). IEEE.
- [59] S. Agarwal, (2014), « Energy Efficient MAC Protocols for Wireless Sensor Network ». International Journal on Computational Sciences & Applications (IJCSA), vol/issue: 4(1), pp. 153-160.
- [60] J. Polastre, J. Hill & D. Culler, (2004). « Versatile low power media access for wireless sensor networks ». In Proceedings of the 2nd international conference on Embedded networked sensor systems. pp. 95-107. ACM.
- [61] A. El-Hoiydi, J. D. Decotignie, C. Enz & E. Le Roux, (2003). « Wisemac, an ultra low power mac protocol for the wisenet wireless sensor network ». In Proceedings of the 1st international conference on Embedded networked sensor systems. pp. 302-303. ACM.
- [62] Z. S. Mahmood, A. A. Hashem, S. A. Hameed, F. Anwar, (2015) « The directional hierarchical AODV routing protocol for wireless mesh networks ». In International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering. IEEE.
- [63] D. O. Castañeda Alcalá, (2015), «Implementación de un modulador-demodulador digital QPSK en banda base a un FPGA como prototipo para un microsatélite». Tesis Ing. U.N.A.M. F.I.
- [64] W. Tomasi (2003). « Comunicaciones digitales en Sistemas de comunicaciones electrónicas». (4ta edición). México: Pearson Educación.
- [65] A. Y. Zúñiga Bolaños, (2018) «Estudio de un sistema cifrado de comunicación inalámbrica usando USRP». Tesis Ing. U.N.A.M. F.I.
- [66] G. Korn & T. Korn, (1968) «Mathematical Handbook for Scientists and Engineers», McGraw-Hill Book Company, Nueva York.
- [67] R. M. Alonso, E. F. Pupo, R. D. Hernández, G. Guillen Nieto, (2015) «Sistema experimental para la caracterización del MER». Revista Telemática Vol. 14 No. 3. ISSN 1729-3804, pp. 30-40.

- [68] Bernard Sklar, (2001). « Digital Communications: Fundamentals and Applications », 2nd ed. USA: Prentice Hall. Pp. 167-236.
- [69] M. Tao, «Principles of Communications. Chapter 8: Digital Modulation Techniques». Shanghai Jiao Tong University.
- [70] S. Haykin, (2001) « Communication Systems ». In John Wiley & Sons, Inc.4th Edition. pp. 349-352
- [71] O. Jaimes, (2017). «Turbo envío: una estrategia de retransmisión rápida para enviar paquetes en redes inalámbricas ad hoc». Trabajo de grado, Maestría en Ingeniería eléctrica-telecomunicaciones, Universidad Nacional Autónoma de México, México.
- [72] J. P. Montero, (2014). «Implementación de un sistema de comunicación basado en Software Radio». Depto. Tecnología Electrónica y de las Comunicaciones, Universidad Autónoma de Madrid. (pp. 28 - 30).
- [73] Ettus, <http://www.ettus.com/all-products/un210-kit/>
- [74] Driver NI USRP, <http://www.ni.com/es-mx/support/downloads/drivers/download.ni-usrp.html>
- [75] J. R. Lajara & J. Pelegrí. (2007) «LabVIEW: Entorno gráfico de programación». Marcombo & Alfaomega.
- [76] National Instrument Support (2013). « Introduction to Software-Defined Radio with LabVIEW and USRP». pp. 7-12. Manual de introducción.
- [77] A. D. Manobanda Dávalos (2015), «Sistema de Análisis y Registro de datos de espectros en el rango de 50Mhz-2.2Ghz a través de Radio Definido por Software». Universidad Técnica de Ambato, Ecuador, pp. 56-60
- [78] H. Moore. (2011). « MATLAB for Engineers ». 3^{ra} Edition. Pearson Education
- [79] Datasheet módulo DIGI Xbee Pro S2C 802.15.4 RF, https://www.digi.com/pdf/ds_xbee-s2c-802-15-4.pdf
- [80] O. A. Soto Sánchez, (2011) «Comparación de la eficiencia volumétrica entre redes inalámbricas WIFI y WIMAX». Tesis Ing. U.N.A.M. F.I.
- [81] A. Goldsmith, (2005) « Wireless Communications». Stanford University, Ed. Cambridge. pp. 38-40.