

# **Simulación de Bluetooth con Matlab®.**



Trabajo de Grado para optar al título de Ingeniero en Electrónica y  
Telecomunicaciones

**Rubén Darío López Muñoz.**

**Fabián Andrés Villamarin Campo.**

Director: Ingeniero en Electrónica y Telecomunicaciones

Víctor Manuel Quintero Florez.

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**

**Departamento de Telecomunicaciones**

**Línea de investigación: Gestión Integrada de Redes, Servicios y  
Arquitecturas de Telecomunicaciones**

Popayán, 2005

A Dios por ser nuestra Roca  
A mi esposa Sandra con todo mi amor, a  
mi hijo Christian y a mis padres Jairo y Gloria.

A mi padre Carlos Alfredo, a mi madre Luz Marina.

A mis hermanos.

A Mechas por compartir conmigo un amor sincero

A Eli y a la pequeña Aleja por su fortaleza.

## AGRADECIMIENTOS

Los autores expresan su agradecimiento a:

Dios por habernos permitido llegar hasta este punto en nuestras vidas.

Víctor Manuel Quintero Florez, Ingeniero Electrónico Telecomunicaciones y director del Trabajo de Grado, por su incansable colaboración y su amistad sincera.

Nuestras familias por su amor y constante apoyo en esta ardua labor.

Natividad Cifuentes por su cariño y su servicio generoso.

Carlos Puerto por su amistad incondicional, consejos y ejemplo de persistencia.

## RESUMEN

La simulación del nivel físico de un sistema de Telecomunicaciones de Bluetooth, tiene como principal objetivo generar una herramienta que pueda ser empleada por la Universidad del Cauca, particularmente por la facultad de Ingeniería Electrónica y Telecomunicaciones, en la enseñanza de la norma IEEE 802.15.1 que define las especificaciones de Bluetooth y las estandariza. Para realizar esta simulación se siguieron los pasos sugeridos por la metodología lineal “Metodología para la simulación de equipos de Telecomunicaciones”. Resaltando la bondades de la herramienta de simulación Simulink®, con la cual se lograron los objetivos trazados mediante el desarrollo asistido de ambiente visual (en la simulación) de los aspectos más importantes del nivel físico de la tecnología Bluetooth.

## **ABSTRACT:**

The simulation of the physical layer of Bluetooth Telecommunications system, must like main objective generate a tool that can be used by the University of the Cauca, particularly by the Electronic Engineering and Telecommunications school, in the education of the norm IEEE 802,15,1 that defines the specifications of Bluetooth and it standardizes them. In order to make Methodology is simulation were followed the steps suggested by the linear methodology "Methodology for the simulation of equipment of Telecommunications". Emphasizing kindness of the tool of Simulink® simulation, with which the objectives drawn up by means of visual environment the attended development (in the simulation) of the most important aspects of the physical level of the Bluetooth technology were obtained, as they are the Frequency Hopping Spread Spectrum(FHSS), the modulation GMSK, the codification among others.

## CONTENIDO

	Pag.
INTRODUCCIÓN	17
1. EL ESTÁNDAR BLUETOOTH BAJO LA NORMA IEEE 802.15.1	19
1.1 NIVEL FÍSICO ( <i>PHYSICAL LAYER</i> , PHY)	23
1.1.1 Requerimientos reglamentarios	24
1.1.2 Bandas de frecuencia y arreglo del canal	24
1.1.3 Características del transmisor	26
1.1.3.1 Características de la modulación	28
1.1.3.2 Tolerancia de RF	32
1.1.4 Características del receptor	33
1.2 ESPECTRO ENSANCHADO ( <i>SPREAD SPECTRUM</i> , SS)	33
1.2.1 Ensanchamiento	33
1.2.2 Espectro ensanchado por secuencia directa (DSSS)	36
1.2.3 Espectro Ensanchado por Salto de Frecuencia (FHSS)	38
1.2.4. Ventajas y desventajas de los dos tipos de espectro ensanchado	40
2. METODOLOGÍA PARA LA SIMULACIÓN DE EQUIPOS DE TELECOMUNICACIONES	42
2.1 FORMULACIÓN DEL PROBLEMA Y PLAN DE ESTUDIOS	44
2.2 RECOLECCIÓN Y PROCESAMIENTO DE DATOS	45
2.3 FORMULACIÓN DE UN MODELO DE SIMULACIÓN	46
2.4 EVALUACIÓN DEL MODELO Y LOS PARÁMETROS ESTIMADOS	47
2.5 PROGRAMACIÓN	49
2.6 VALIDACIÓN DEL PROGRAMA	52
2.7 DISEÑO DE EXPERIMENTOS	53

2.8 ANÁLISIS DE LOS DATOS OBTENIDOS CON LA SIMULACIÓN	53
2.9 DOCUMENTO E IMPLEMENTACIÓN DE RESULTADOS	54
3. MODELO CONCEPTUAL DE BLUETOOTH	56
3.1 CONTENIDO TEÓRICO	57
3.1.1 Extracto de la norma IEEE 802.15.1 WPAN tecnología Bluetooth	57
3.1.2 Definición de canal	58
3.1.3 Definición de paquete	59
3.1.4 Definición de enlace físico	60
3.1.5 Transmisión de datos	62
3.1.6 Reconocimiento de los datos en el transmisor	64
3.2 MODELOS CONCEPTUALES	65
3.2.1 Identificación de conceptos dentro de la tecnología Bluetooth	68
3.2.2 Identificación de asociaciones del sistema Bluetooth	69
3.2.3 Identificación de Atributos	70
3.2.4 Fundamentación del modelo conceptual del sistema Bluetooth	72
3.3 DIAGRAMA CONCEPTUAL DE LAS CAPAS BAJAS DEL SISTEMA BLUETOOTH.	80
3.3.1 Diagrama completo y diagrama alternativo de secuencias del sistema Bluetooth.	81
3.3.1.1 Modulo Transmisor ( TX )	81
3.3.1.2 Modulo Receptor ( RX )	82
3.3.1.3 Diagrama General (Incluye tipos de señal)	82
3.3.2 Diagrama de secuencias Bluetooth basado en las asociaciones	83
3.3.3 Procedimientos del Diagrama Secuencial respecto al Modelo Conceptual	84
4. ANÁLISIS DE LA SIMULACIÓN	86
4.1 DATOS	88
4.2 CODIFICACIÓN	91
4.3 ENTRAMADO	100
4.4 SALTO DE FRECUENCIA	102
4.5 MODULACIÓN GFSK Y MEZCLA CON CANALES DE FI	103

4.6 DEMODULACIÓN DE FI, DESENTRAMADO Y DECODIFICACIÓN	106
5. APLICACIÓN DE LA METODOLOGÍA DE SIMULACIÓN ADAPTADA A MATLAB®.	107
5.1 CONCEPTOS	108
5.1.1 Sistema	108
5.1.2 Simulación	108
5.2 APLICACIÓN DE LA METODOLOGÍA DE SISTEMAS DE TELECOMUNICACIONES ADAPTADA A MATLAB®	109
5.2.1 Formulación del problema y plan de estudios	110
5.2.2 Recolección y procesamiento de datos	112
5.2.3 Formulación del modelo de simulación del sistema de enlace transmisor – receptor Bluetooth	112
5.2.4 Evaluación del Modelo y sus parámetros estimados	113
5.3 PROGRAMACIÓN DEL SISTEMA TRANSMISOR – RECEPTOR BLUETOOTH BAJO SIMULINK®	119
5.3.1 Canal	120
5.3.2 Bloque de Modulación de Frecuencia GFSK	122
5.3.3 Entramado	124
5.3.4 Codificación	125
5.3.5 Datos	128
5.3.6 Demodulador	131
5.3.7 Desentramado	133
5.3.8 Decodificador	135
5.4 VALIDACIÓN DEL PROGRAMA	136
5.5 MODELO TOTAL DE LA SIMULACIÓN	145
6. CONCLUSIONES Y RECOMENDACIONES	147
7. BIBLIOGRAFÍA Y REFERENCIAS	150



## LISTA DE FIGURAS

	Pag.
Figura 1. Pila del protocolo Bluetooth.	20
Figura 2. Comparación de Bluetooth con el modelo de referencia OSI.	21
Figura 3. Relaciones de interfaz de nivel físico.	24
Figura 4. Modulación real de transmisión.	29
Figura 5. FSK.	30
Figura 6. Subsistema FSK.	31
Figura 7. Sistema GMSK.	32
Figura 8. Espectro ensanchado	34
Figura 9. Saltos de frecuencia.	39
Figura 10. Vías de desarrollo para un trabajo de simulación.	43
Figura 11. Intervalos de tiempo.	58
Figura 12. Estructura de un paquete.	60
Figura 13. Diagrama de tres llamadas de voz simultáneas.	62
Figura 14. Registro de tiempo de paquetes DM1 y NULL.	64
Figura 15. Diagrama de estados.	66
Figura 16. Representación señal.	72
Figura 17. Representación control errores.	73
Figura 18. Representación ensamble.	74
Figura 19. Representación modulación.	75
Figura 20. Representación demodulación.	76
Figura 21. Representación canal.	76
Figura 22. Representación secuencia.	77
Figura 23. Representación espectro.	77

Figura 24. Representación banda.	78
Figura 25. Representación del intervalo.	78
Figura 26. Representación transmisor.	79
Figura 27. Representación Receptor.	79
Figura 28. Comunicación entre el transmisor y el receptor.	81
Figura 29. Especificación del transmisor genérico.	81
Figura 30. Especificación del receptor genérico.	82
Figura 31. Modelo formal de conceptos completo.	82
Figura 32. Diagrama de secuencias del sistema TX y RX Bluetooth.	84
Figura 33. Modelo unificado.	85
Figura 34. Bloque generador de tren binario aleatorio	121
Figura 35. Generador Bernoulli para datos.	122
Figura 36. Codificador Hamming.	123
Figura 37. Configuración del Codificador Hamming.	124
Figura 38. Buffer de Simulink®.	125
Figura 39. Configuración del buffer para entramado.	125
Figura 40. Modelo de Modulador GFSK.	127
Figura 41. Configuración del Bloque M - FSK.	128
Figura 42. Configuración del bloque producto.	129
Figura 43. Bloque de Simulink®.	129
Figura 44. Configuración del canal de transmisión.	131
Figura 45. Bloque de demodulación GMSK.	132
Figura 46. Bloque función matemática.	133
Figura 47. Configuración de la función matemática.	134
Figura 48. Bloque de desentramado.	134
Figura 49. Configuración del buffer del receptor.	135
Figura 50. Configuración del retardo.	136
Figura 51. Decodificador Hamming.	136
Figura 52. Configuración del decodificador Hamming.	137
Figura 53. Generador de datos.	138
Figura 54. Codificador Hamming.	139

Figura 55. Bloque de ensamble de paquetes.	140
Figura 56. Bloque GMSK y saltos de frecuencia.	141
Figura 57. Bloque del canal de TX.	141
Figura 58. Bloque generador de salto.	142
Figura 59. Bloque demodulador.	143
Figura 60. Bloque desensamble de paquetes.	144
Figura 61. Bloque decodificador Hamming.	145
Figura 62. BER	146
Figura 63. Diagrama total de la simulación.	147

## LISTA DE TABLAS

	Pag.
Tabla 1. Bandas de frecuencia de operación.	25
Tabla 2. Bandas de guarda.	26
Tabla 3. Clases de potencia.	27
Tabla 4. Pasos de una simulación	42
Tabla 5. Atributos del concepto señal	73
Tabla 6. Atributos del concepto control de errores	74
Tabla 7. Atributos del concepto ensamble	75
Tabla 8. Atributos del concepto modulación	75
Tabla 9. Atributos del concepto demodulación	76
Tabla 10. Atributos del concepto canal	76
Tabla 11. Atributos del concepto secuencia	77
Tabla 12. Atributos del concepto espectro	78
Tabla 13. Atributos del concepto banda	78
Tabla 14. Atributos del concepto intervalo	79
Tabla 15. Atributos del concepto transmisor	79
Tabla 16. Atributos del concepto receptor	80

## ACRONIMOS

<b>AC</b>	Corriente Alterna	<i>Alternating Current</i>
<b>ACI</b>	Interferencia de Canal Adyacente	<i>Adjacent Channel interferente</i>
<b>ACK</b>	Confirmar	<i>Acknowledge</i>
<b>ACL</b>	Asíncrono no Orientada a la Conexión	<i>Asynchronous Conecction Less</i>
<b>ARQ</b>	Solicitud de Repetición Automática	<i>Automatic Repeat Request Address</i>
<b>BB</b>	Banda Base	<i>Baseband</i>
<b>BER</b>	Tasa de Error de Bit	<i>Bit Error Rate</i>
<b>BNEP</b>	Protocolo de Encapsulación de Red Bluetooth	<i>Bluetooth Network Encapsulation Protocol</i>
<b>BPSK</b>	Modulación de Cambio de Fase Binaria	<i>Biphase Shift Keying</i>
<b>CDMA</b>	Acceso Múltiple por División de Código	<i>Code Division Multiple Access</i>
<b>CRC</b>	Codificación de Redundancia Cíclica	<i>Cyclic Redundancy Code</i>
<b>CVSD</b>	Modulación Delta con Pendiente de Variable Continua	<i>Continuously Variable Slope Delta Modulation</i>
<b>DH</b>	Paquetes de Datos Largos	<i>Data High</i>
<b>DSP</b>	Procesamiento Digital de Señales	<i>Digital Signal Procesing</i>
<b>DSSS</b>	Espectro Ensanchado de Secuencia Directa	<i>Direct Secuence Spread Spectrum</i>
<b>DV</b>	Paquetes de Voz y Datos	<i>Data - Voice</i>
<b>ETC</b>	Condiciones de Prueba Extremas	<i>Extremal Test Conditions</i>

<b>ETSI</b>	Instituto de Estándares de Telecomunicaciones Europeo	<i>European Telecommunications Standards</i>
<b>FCC</b>	Comisión de Comunicaciones Federales	<i>Federal Communications Comisión</i>
<b>FH</b>	Salto de frecuencia	<i>Frequency hopping.</i>
<b>FHSS</b>	Espectro Ensanchado por Salto de Frecuencia	<i>Frequency Hopping Spread Spectrum</i>
<b>FSK</b>	Modulación por Cambio de Frecuencia	<i>Frequency Shift Keying</i>
<b>GFSK</b>	Cambio de Frecuencia Gaussiana	<i>Gaussian Frequency Shift Keying</i>
<b>GHz</b>	Gigahercio	<i>Gigahertz</i>
<b>HCI</b>	Interfaz Controladora de Host	<i>Host Controller Interface</i>
<b>IEEE</b>	Instituto de Ingenieros en Electricidad y Electrónica	<i>Institute of Electrical and Electronics Engineers</i>
<b>ISM</b>	Médico, Científica e Industrial	<i>Industrial, Scientific, Medical</i>
<b>Kbps</b>	Kilobits por segundo	<i>Kilobits per second</i>
<b>KHz</b>	Kilohercios	<i>Kilohertz</i>
<b>L2CAP</b>	Protocolo de Adaptación y Control de Enlace Lógico	<i>Logical Link Control and Adaptation Protocol</i>
<b>LAN</b>	Red De Área Local	<i>Local Area Network</i>
<b>LC</b>	Controlador de Enlace	<i>Link Controller</i>
<b>LM</b>	Gestor de Enlace	<i>Link Manager</i>
<b>LMP</b>	Protocolo Gestor de Enlace	<i>Link Manager Protocol</i>
<b>MAC</b>	Control de Acceso al Medio	<i>Medium Access Control</i>
<b>MSB</b>	Bit Más Significativo	<i>Most Significant Bit</i>
<b>Mw</b>	Miliwatio	<i>Milliwatt</i>
<b>NTC</b>	Condición de Prueba Nominal	<i>Nominal Test Conditions</i>
<b>OBEX</b>	Protocolo de intercambio de objeto	<i>Object Exchange Protocol</i>

<b>OSI</b>	Sistemas de Interconexión Abierta	<i>Open System Interconnection</i>
<b>ppm</b>	Pulsos por minuto	<i>pulses per minute</i>
<b>PPP</b>	Protocolo Punto a Punto	<i>Per to Per Protocol</i>
<b>PRBS</b>	Secuencia de Bits Pseudo Aleatoria	<i>Pseudo Random Bit Sequence,</i>
<b>QPSK</b>	Modulación de Cambio de Fase de Cuadratura	<i>Quadrature phase shift keying</i>
<b>RF</b>	Radiofrecuencia	<i>Radio Frequency</i>
<b>RFCOMM</b>	Comunicación por Radiofrecuencia	<i>Radio Frequency comunicat</i>
<b>RSSI</b>	Medición de Intensidad de la Señal del Receptor	<i>Receiver Signal Strength Indication</i>
<b>RX</b>	Receptor	<i>Receiver</i>
<b>SAP</b>	Punto de Acceso de Servicio	<i>Services Access Points</i>
<b>SCO</b>	Enlace Sincrono Orientado a la Conexión	<i>Synchronous Connection Oriented</i>
<b>SDP</b>	Protocolo de Descubrimiento de Servicios	<i>Service Discovery Protocol</i>
<b>SIG</b>	Grupo de Interés Especial	<i>Special Interest Group</i>
<b>SNR</b>	Razón Señal a Ruido	<i>Signal Noise Ratio</i>
<b>SS</b>	Espectro Ensanchado	<i>spread spectrum</i>
<b>TCP/IP</b>	Protocolo de Control de Transmisión	<i>Transmission Control Protocol / Internet Protocol</i>
<b>TDD</b>	Duplexación por División de Tiempo	<i>Time Division Duplex</i>
<b>TX</b>	Transmisor	<i>Transmitter</i>
<b>WAP</b>	Protocolo de Acceso Inalámbrico	<i>Wireless Access Protocolo</i>
<b>WPAN</b>	Redes Inalámbricas de Área Personal	<i>Wireless Personal Area Network</i>

## INTRODUCCIÓN

En 1994 Ericsson inició un estudio para investigar la viabilidad de una interfaz vía radio, de bajo costo y bajo consumo, para la interconexión entre teléfonos móviles y otros accesorios con la intención de eliminar cables entre dispositivos. El estudio partió de un largo proyecto que investigaba sobre unos comunicadores conectados a una red celular, hasta que se llegó a un enlace de radio de corto alcance, llamado *MC link*. Conforme éste proyecto avanzaba se vislumbró que éste tipo de enlace podía ser utilizado ampliamente en un gran número de aplicaciones, ya que tenía como principal virtud el que se basaba en un chip de radio relativamente económico.

A comienzos de 1997, según avanzaba el proyecto *MC link*, Ericsson fue despertando el interés de otros fabricantes de equipos portátiles. En seguida se vio claramente que para que el sistema tuviera éxito, un gran número de equipos deberían estar equipados con esta tecnología. Esto fue lo que originó a principios de 1998, la creación de un grupo de interés especial (*Special Interest Group*, SIG), formado por 5 promotores que fueron: Ericsson, Nokia, IBM, Toshiba e Intel [1].



Se pretende con este trabajo de grado tener un mayor acercamiento a las nuevas tecnologías de radio de corto alcance, a los esquemas avanzados de modulación y de codificación y a las técnicas de acceso basadas en espectro ensanchado mediante la simulación, soportados por un potente software matemático como lo es Matlab® y sus conjuntos de herramientas aplicados a la comunicación y al procesamiento de señales. Este tipo de simulaciones generará productos que beneficiarán tanto a profesores como estudiantes de la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca, mejorando los procesos de enseñanza y aprendizaje de las asignaturas relacionadas con las comunicaciones móviles e inalámbricas.

El desarrollo de este trabajo se fundamentó en cuatro aspectos principalmente, la consecución de la información, la elección de una metodología adecuada para el desarrollo de la simulación, la orientación de esta metodología hacia la tecnología Bluetooth y finalmente plasmar los aspectos anteriormente mencionados en la simulación en si, empleando la herramienta de Matlab® particularmente Simulink®.

## 1. EL ESTÁNDAR BLUETOOTH BAJO LA NORMA IEEE 802.15.1

La norma IEEE 802.15.1 (Bluetooth) pertenece a la serie de normas del Instituto de Ingenieros Eléctricos y Electrónicos (*Institute of Electrical and Electronics Engineers, IEEE*) relacionadas con Redes Inalámbricas de Área Personal (*Wireless Personal Area Network, WPAN*). Este capítulo constituye una referencia del nivel físico de acuerdo con el modelo de referencia de Interconexión de Sistemas Abiertos (Open System Interconnection, OSI) y para una mejor interpretación de lo que se expone se incluye la descripción de niveles que están por encima del nivel físico. Se realiza una comparación de los conceptos tecnológicos utilizados actualmente en la técnica de Espectro Ensanchado (*spread spectrum, SS*) específicamente entre Secuencia Directa (Direct Sequence, DS) y Saltos de Frecuencia (*Frequency Hopping, FH*).

La pila de protocolos Bluetooth incluye protocolos específicos tales como Protocolo Gestor de Enlace (*Link Manager Protocol, LMP*), Protocolo de Adaptación y Control de Enlace Lógico (*Logical Link Control and Adaptation Protocol, L2CAP*) y los protocolos no específicos. Estos últimos incluyen el Protocolo de Intercambio de Objeto (*Object Exchange Protocol, OBEX*), el Protocolo Punto a Punto (*Per to Per Protocol, PPP*), el Protocolo de Acceso

Inalámbrico (*Wireless Access Protocol*, WAP), Protocolo de Encapsulación de Red Bluetooth (*Bluetooth Network Encapsulation Protocol*, BNEP), Interfaz Controladora de Host (*Host Controller Interface*, HCI) etc. Para aclarar esto se muestra a continuación en la figura 1 una pila del protocolo Bluetooth.

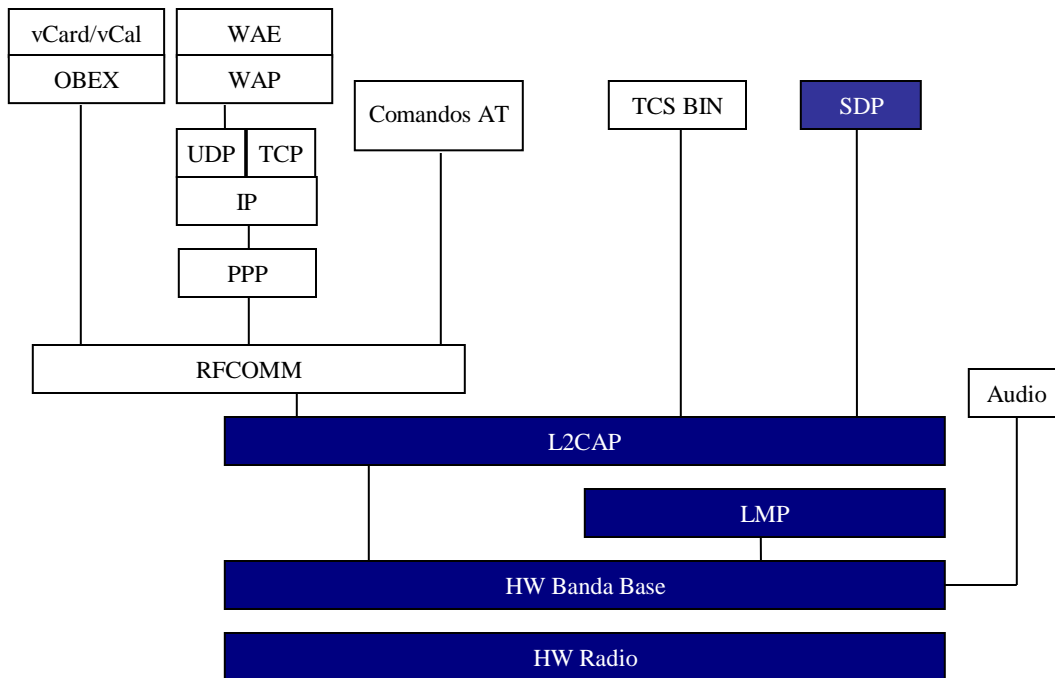


Figura 1. Pila del protocolo Bluetooth.

Los elementos oscuros son los que obligatoriamente debe contener un dispositivo Bluetooth, la parte inferior es la del hardware del sistema, que incluye un módulo de banda base, así como un módulo de radio y una antena (que no está representada en la figura). Además, para que dos dispositivos Bluetooth puedan comunicarse es necesario un módulo de software Gestor de Enlace (*Link Manager*, LM), que por lo general corre en un microprocesador dedicado. Los LM de dos dispositivos Bluetooth se comunican entre sí mediante el protocolo LMP. También es necesario el protocolo L2CAP, que ofrece servicios de datos a los

protocolos de niveles superiores. Por último, el Protocolo de Descubrimiento de Servicios (*Service Discovery Protocol, SDP*) el cual sirve para obtener información sobre otros dispositivos Bluetooth, con el propósito de comunicarse con ellos y utilizar los servicios que puedan ofrecer.

Estos protocolos son el núcleo de Bluetooth y todos los dispositivos Bluetooth deben implementarlos. El resto de protocolos entre L2CAP y la propia aplicación sólo serán necesarios si lo requiere la aplicación concreta. La mayor o menor complejidad de la pila de protocolos para una aplicación dada vendrá determinada por los requerimientos de ésta.

A continuación se puede apreciar en la figura 2, un paralelo entre la pila de protocolos de Bluetooth con el modelo de referencia OSI estandarizado

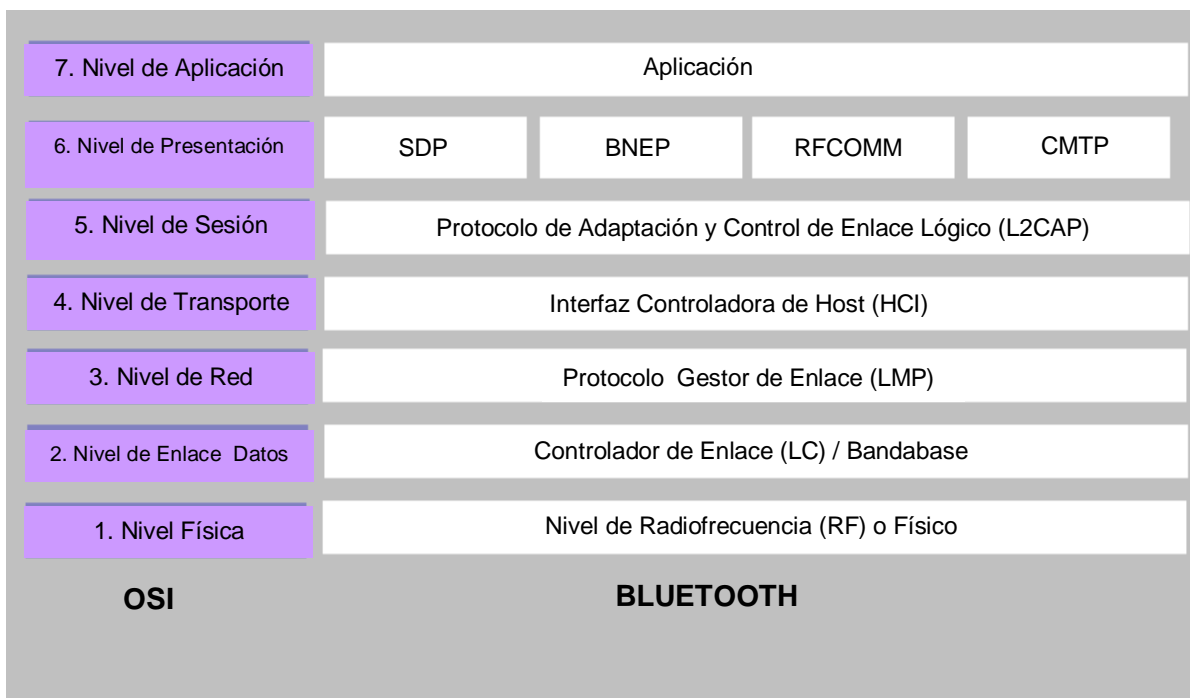


Figura 2. Comparación de Bluetooth con el modelo de referencia OSI.

La función de cada nivel se resume como sigue:

- Físico: Modula y demodula los datos para la transmisión y recepción a través del aire.
- Banda Base y Controlador de Enlace (*Link Control*, LC): Controlan los enlaces físicos vía radio, ensamblando paquetes y generando el salto de frecuencia.
- Gestor de Enlace (LM): Controla y configura los enlaces con otros dispositivos.
- Interfaz Controladora de Host (HCI): Lleva las comunicaciones entre un módulo Bluetooth y un computador, permitiéndole a este último acceder a las capacidades físicas del módulo.
- Protocolo de Adaptación y Enlace Lógico (L2CAP): Distribuye y acondiciona el tamaño de paquetes para los niveles superiores.
- RFCOMM: Protocolo de Emulación de Cable Serial, el nombre RFCOMM viene de Radiofrecuencia (RF), orientada a la emulación de los puertos seriales COM de un computador. RFCOMM emula completamente la comunicación serial RS-232 de nueve pines sobre un canal L2CAP.
- Protocolo de Descubrimiento de Servicio (SDP): Permite descubrir los servicios brindados por otros dispositivos.

- OBEX, WAP: Suministra otros protocolos de comunicación a los niveles altos, si estos son requeridos.

## 1.1 NIVEL FÍSICO (*PHYSICAL LAYER, PHY*)

La figura 3 muestra la relación entre la pila del protocolo Bluetooth y el nivel físico. El nivel físico es el primero de 7 niveles del modelo OSI y es el responsable de transmitir los bits haciendo uso del sistema de radio a través del aire. La descripción de este nivel en cuanto a la parte de radio, está limitada a lo siguiente:

- Recibir un cadena de bits desde el subnivel Control de Acceso al Medio (*Medium Access Control, MAC*) y transmitir la cadena de bits vía ondas de radio a una estación asociada.
- Recibir ondas de radio desde una estación asociada y convertir estas a una cadena de bits para pasarla al subnivel MAC.

Esto refleja el alcance limitado de la porción del nivel físico de radio de la arquitectura IEEE 802.15.1. Los bits y las ondas de radio son transmitidos, pero este nivel no hace ninguna interpretación de ellos.

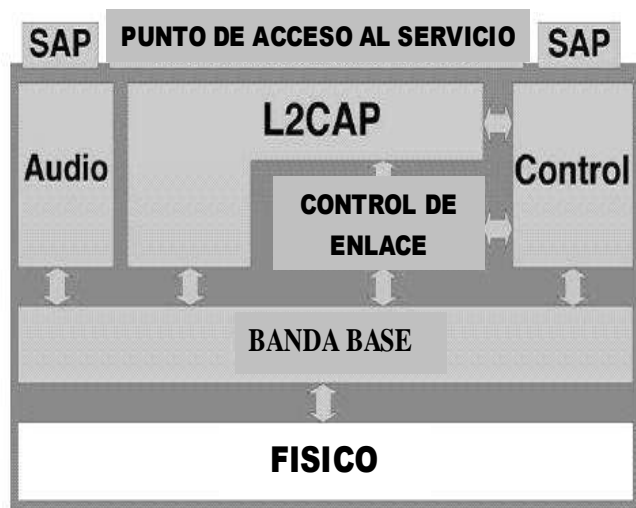


Figura 3. Relaciones de interfaz de nivel físico.

**SAP: Punto de Acceso al Servicio (Service Access Point)**

**1.1.1 Requerimientos reglamentarios** El transceptor (llamado también Transmisor – Receptor) de Bluetooth opera en la banda ISM. El transceptor Bluetooth cumplirá los requisitos declarados bajo las condiciones de funcionamiento que se especificarán en los numerales 1.5 y 1.6. Estas condiciones están basadas en las regulaciones establecidas para Europa, Japón y Norte América.

**1.1.2 Bandas de frecuencia y arreglo del canal** Bluetooth opera en la banda de frecuencia de 2,4 GHz o banda ISM. En la mayoría de países alrededor del mundo, el rango de esta banda de frecuencia va desde los 2.400 MHz a los 2.483,5 MHz. Algunos países, sin embargo, tienen limitaciones nacionales en el rango de frecuencias. Para cumplir con estas limitaciones nacionales, algoritmos

especiales para saltos de frecuencia han sido especificados por esos países, con el fin de reducir su banda. Es de anotarse que los productos que implementan la banda de frecuencia reducida no trabajarán con los que emplean la banda completa. Los productos que implementan la banda reducida, por tanto, se consideran como versiones locales para un solo mercado (ver Tabla 1).

Tabla 1. Bandas de frecuencia de operación.

<b>Geografía</b>	<b>Rango Regulatorio (GHz)</b>	<b>Canales de Radiofrecuencia</b>
Estados Unidos, Europa y la mayoría de los otros países	2,400 – 2,4835	$F = 2.402 + k \text{ MHz}, k = 0, \dots, 78$
Francia	2,4465 – 2,4835	$f = 2.454 + k \text{ MHz}, k = 0, \dots, 22$

El ancho de banda por canal es de 1 MHz. Durante la comunicación se conmuta o se realiza el salto a través de 79 o 23 canales de Radiofrecuencia (Radiofrequency, RF) dependiendo de la secuencia pseudoaleatoria utilizada. La secuencia de salto es única para la pico-red y esta determinada por la dirección del dispositivo maestro de Bluetooth. El sincronismo en la secuencia de salto esta determinada por el reloj del maestro de Bluetooth. El canal está dividido en intervalos de tiempo donde cada intervalo corresponde a una frecuencia de salto. Estas frecuencias de saltos corresponden a saltos consecutivos a diferentes frecuencias. La tasa de salto nominal es de 1600 saltos por segundo. Con el propósito de cumplir con las regulaciones en cada país, se utiliza una banda de guarda. Los países tienen un rango de frecuencias definido donde aparecen estas bandas de guarda (ver Tabla 2).



Tabla 2. Bandas de guarda.

Geografía	Banda de Guarda Inferior (MHz)	Banda de Guarda Superior (MHz)
Estados Unidos, Europa y la mayoría de los otros países	2	3,5

**1.1.3 Características del transmisor** Los requerimientos están dados como niveles de potencia en el conector de la antena del equipo. Si el equipo no tiene un conector, se asume una antena de referencia con 0 dBi de ganancia. Esto trae como resultado la dificultad en la exactitud de la medida.

Se utilizan antenas de ganancia direccional superior a 0 dBi, la potencia de transmisión satisface las normas del Instituto de Estándares de Telecomunicaciones Europeo (*European Telecommunications Standards institute*, ETSI) ETS 300-328 o de la Comisión Federal de Comunicaciones (*Federal Communications Commission*, FCC) FCC's CFR47, Parte 15.

Se requiere control de potencia para equipos de potencia clase 1. El equipo de potencia clase 1 se caracteriza porque su potencia de salida se encuentra definida entre los rangos de 100 mW (20 dBm) como su valor máximo y 1 mW (0 dBm) como su valor mínimo, el equipo Clase 1 es capaz de controlar su potencia a transmitir por debajo de 4 dBm o menos. El control de potencia se utiliza para limitar la potencia transmitida sobre 0 dBm. La capacidad de control de potencia por debajo de 0 dBm es opcional y podría ser utilizada para optimizar el consumo de energía y el nivel de interferencia global. Los cambios de potencia deben

formar una secuencia monotónica con un cambio máximo de 8 dB y un cambio mínimo de 2 dB.

Los equipos están divididos dentro de tres clases de potencia, así:

Tabla 3. Clases de potencia.

Clase de Potencia	Potencia máxima de salida $P_{\text{omax}}$	Potencia de salida nominal	Potencia mínima de salida <sup>a</sup>	control de potencia
1	100 mW (20 dBm)	N/A	1mW ( 0 dBm )	$P_{\text{min}} < +4$ dBm hasta $P_{\text{max}}$ . Opcional: $P_{\text{min}}^b$ hasta $P_{\text{max}}$ .
2	22,5 mW (4dBm)	1mW ( 0 dBm )	0,25 mW (-6 dBm)	Opcional: $P_{\text{min}}^b$ hasta $P_{\text{max}}$ .
3	1 mW (0 dBm)	N/A	N/A	Opcional: $P_{\text{min}}^b$ hasta $P_{\text{max}}$

<sup>a</sup> se refiere a la mínima potencia de salida en configuración de máxima potencia

<sup>b</sup> se refiere al limite de potencia baja  $P_{\text{min}} < -30$ dBm sugerida , pero no es obligatorio, y puede escogerse de acuerdo con las necesidades de la aplicación.

El equipo con capacidad para controlar la potencia, optimiza la potencia de salida en un enlace con comandos del LMP. Esto se hace mediante Medición de Intensidad de la Señal en el Receptor (*Receiver Signal Strength Indication, RSSI*) y retornando un reporte si la potencia debe ser incrementada o decrementada.

La potencia clase 1 no será utilizada para envío de paquetes de un dispositivo a otro si el lado de recepción de una conexión no soporta la mensajería necesaria para control de potencia del lado que emite (ej, las mediciones RSSI y los mensajes relacionados). En este caso, el transmisor debería cumplir con las reglas de un transmisor clase 2 o clase 3. Adicionalmente, si un dispositivo clase 1 está buscando en proximidad cercana (*paging*) a otro dispositivo, la potencia de

entrada podría ser incrementada hasta el estado requerido. Esto puede causar al dispositivo que escucha una falla de respuesta. Un equipo que realiza búsqueda o exploración se hará de acuerdo con la clase de potencia 2 o 3. [2], [3]

**1.1.3.1 Características de la modulación** Bluetooth utiliza una Modulación por Cambio de Frecuencia Gaussiana (*Gaussian Frequency Shift Keying*, GFSK) con un tiempo de ancho de banda  $BT = 0,5$ . El índice de modulación debe estar entre 0,28 y 0,35. Un 1 binario se representa por una desviación de frecuencia positiva y un 0 binario como una desviación de frecuencia negativa con respecto a la frecuencia transmitida.

Por cada canal transmitido, la desviación de frecuencia mínima ( $F_{min} =$  la menor de  $\{F_{min+}, F_{min-}\}$ ) que corresponde a la secuencia 1010 no debe ser menor de  $\pm 80\%$  de la desviación de frecuencia ( $f_d$ ) que corresponde a la secuencia 00001111. Adicionalmente, la desviación mínima no debe ser menor de 115 KHz. El dato transmitido tiene una tasa de símbolo de 1Mss (Mega símbolo por segundo).

El error de cruce por cero es la diferencia entre el periodo de símbolo ideal y el tiempo de cruce medido. Este debe ser menor de  $\pm 0,125$  de periodo de un símbolo. La desviación de frecuencia máxima debe estar entre 140 KHz y 175KHz.

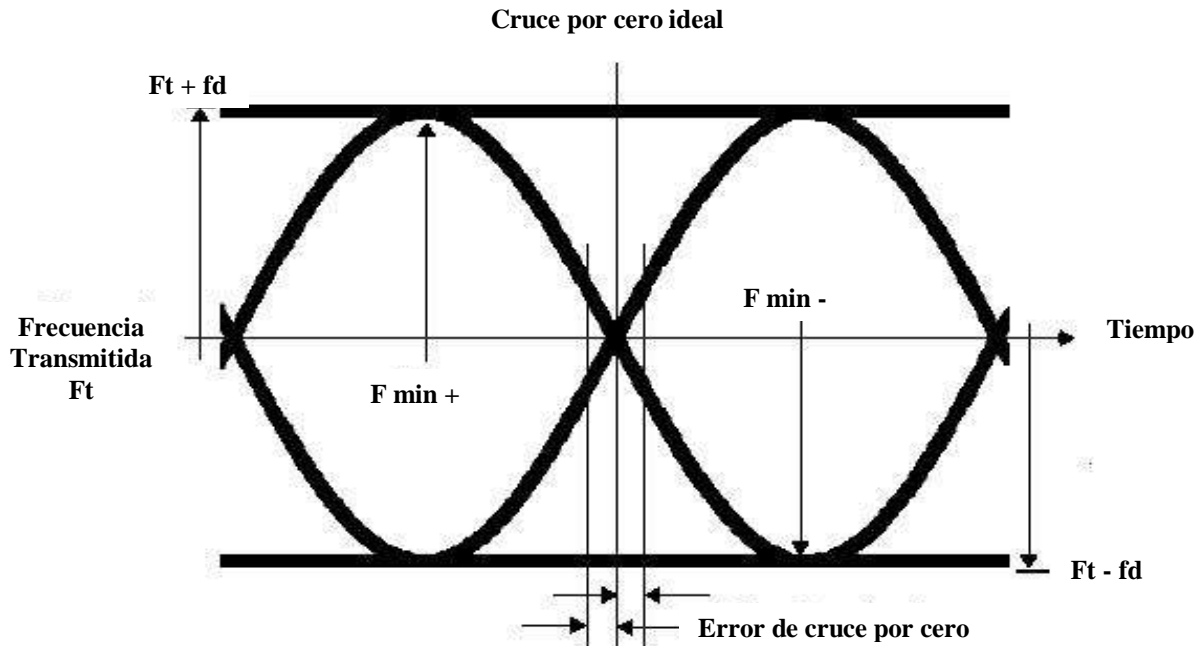


Figura 4. Modulación real de transmisión.

GFSK o GMSK está basado en Modulación por Cambio de Frecuencia (*Frequency Shift Keying*, FSK), por lo cual se explicará primero este método de modulación.

- **Modulación por Cambio de Frecuencia, FSK** En la modulación por desplazamiento de frecuencia (FSK) la presencia de un tono de frecuencia representa un "1" y la presencia de otra frecuencia representa un "0" denominadas frecuencia de marca y espacio respectivamente. Para permitir transmisión simultánea en dos sentidos, dos pares de frecuencias son utilizadas comúnmente. La ventaja de este tipo de modulación es la robustez ante la presencia de ruido e interferencia. La desventaja es la complejidad del sistema (mayor costo) y el consumo de energía permanente durante la transmisión. Este tipo de modulación se utiliza en los módems de baja velocidad

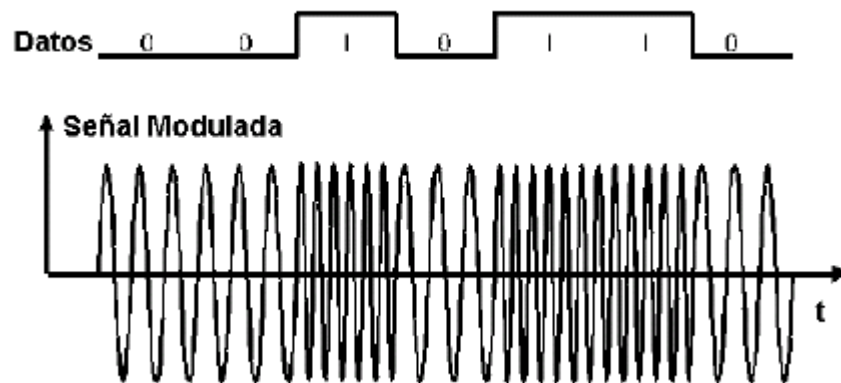


Figura 5. FSK.

La señal,  $s(t)$  de una señal FSK puede ser representada por la siguiente ecuación:

$$s(t) = \sqrt{\frac{2E_s}{T}} \cos\left(2\pi f_c t + \frac{a_n h(t - nT)}{T} + \theta\right) \quad (1)$$

$$nT \leq t \leq (n+1)T$$

Donde:

$E_s$ : Es la potencia de la señal

$T$ : Es la duración del símbolo (1E-6 para Bluetooth)

$f_c$ : Es la frecuencia portadora (~ 2,4 GHz para Bluetooth)

$h$ : Es el índice de modulación (para Bluetooth es de 0,32)

$a_n$ : Es el n-esimo bit de datos

$\theta$ : Es la constante de cambio de fase.

En FSK, la señal  $s(t)$  esencialmente conmuta entre dos frecuencias. El índice de modulación  $h$  determina la separación entre estas dos frecuencias. Para

Bluetooth el índice de modulación debe tener un valor de 0,32 con una variación del 1%. El bit de datos  $a_n$  tiene un valor de -1 o +1.

El siguiente diagrama de bloques en Simulink® puede ser empleado para implementar FSK:

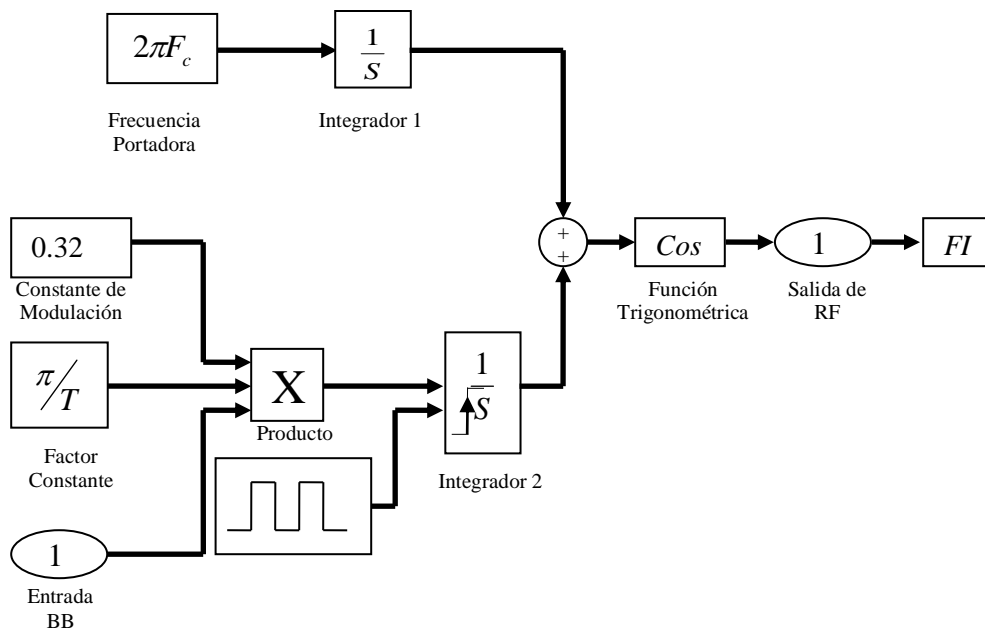


Figura 6. Subsistema FSK.

- **Modulación por Cambio de Frecuencia Gaussiana, GFSK o GMSK** GMSK es simplemente FSK pero la entrada se pasa primero a través de un filtro Gaussiano, este filtro es un filtro pasa bajos cuyo objetivo es eliminar los lóbulos laterales de la señal banda base.

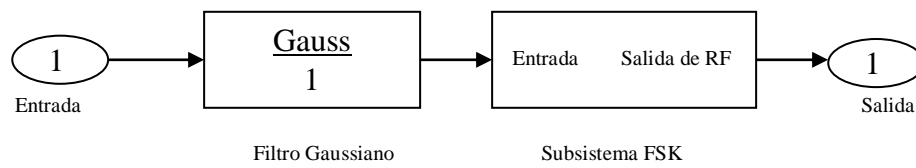


Figura 7. Sistema GMSK.

El filtro Gaussiano está dado por la siguiente expresión matemática:

$$g(t) = \frac{1}{\sqrt{2\pi}\sigma T} \exp\left(\frac{-t^2}{2\sigma^2 T^2}\right) \quad (2)$$

$$\sigma = \frac{\sqrt{\ln(2)}}{2\pi BT} \quad (3)$$

Donde:

**BT:** Controla el ancho de banda del filtro Gaussiano. Para Bluetooth,  $BT = 0,5 \pm 1\%$ .

**1.1.3.2 Tolerancia de RF** La frecuencia central del transmisor ( $F_c$ ) debe estar dentro del parámetro permitido el cual tiene una desviación máxima de  $\pm 75$  KHz desde  $F_c$ . La exactitud de la frecuencia inicial está definida como la precisión de la frecuencia antes que cualquier información sea transmitida.

**1.1.4 Características del receptor** Para medir la Tasa de Error de Bit (*Bit Error Rate*, BER), el equipo debe tener una facilidad de realimentación. El equipo envía de vuelta la información decodificada. El nivel de referencia requerido es de – 70 dBm.

## **1.2 ESPECTRO ENSANCHADO (*SPREAD SPECTRUM, SS*)**

La tecnología inalámbrica Bluetooth utiliza una técnica de codificación llamada espectro ensanchado o expansión de espectro, un método de comunicaciones inalámbricas que toma una señal de banda estrecha y la expande en una porción más amplia de la banda de frecuencias de radio disponible. Entre otras ventajas, la señal resultante es muy resistente a interferencias y más segura contra interceptaciones. La misma tecnología se utiliza en teléfonos celulares y Redes de Área Local Inalámbrica (*Wireless Local Area Network*, WLAN). Además, muchos servicios celulares utilizan Acceso Múltiple por División de Código (*Code Division Multiple Access*, CDMA), una técnica de modulación y acceso que se basa en el concepto de espectro ensanchado.

**1.2.1 Ensanchamiento** El espectro ensanchado es una técnica de codificación digital en la que se toma una señal de banda estrecha y se “expande” sobre un espectro de frecuencias como se indica en la figura 8. La operación de



codificación aumenta el número de bits transmitidos y expande el ancho de banda utilizado. Utilizando el mismo código de expansión o ensanchamiento que el transmisor, el receptor correlaciona y devuelve la señal expandida a su forma original. El resultado es una tecnología de transmisión de datos inalámbrica muy robusta, que ofrece unas ventajas considerables de funcionamiento respecto a sistemas convencionales de banda estrecha.

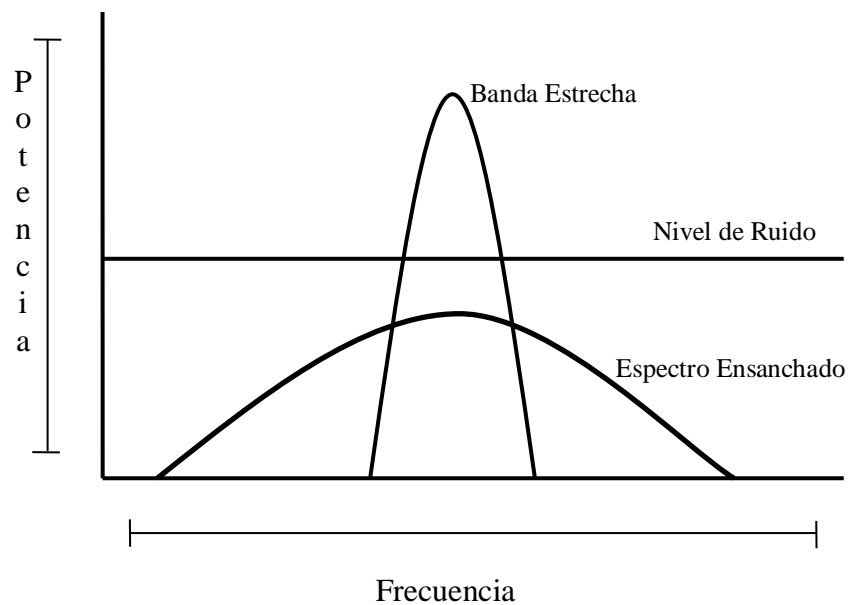


Figura 8. Espectro ensanchado

Una de las ventajas del espectro ensanchado es que la señal ensanchada tiene una densidad de potencia mucho menor. Esta baja densidad de potencia, ensanchada sobre el ancho de banda del transmisor, proporciona resistencia a una serie de condiciones que pueden afectar a los sistemas de banda estrecha, incluyendo:

- Interferencia. Condición en la que se interrumpe una transmisión a causas de fuentes externas, como el ruido emitido por dispositivos electromagnéticos, o a causa de fuentes internas como la diafonía.
- Interferencia intencionada. Condición en la que una señal más fuerte se sobrepone a una señal más débil, causando una interrupción en la comunicación de esta última.
- Trayecto múltiple. Condición en la que la señal original se distorsiona tras recorrer distintos trayectos y llegar a receptor con una fase diferente.
- Interceptación. Condición en la que usuarios no autorizados capturan señales intentando determinar su contenido.

Los sistemas de radio convencionales de banda estrecha transmiten y reciben en una frecuencia específica que es solo lo suficientemente amplia como para transmitir la información, ya sea voz o datos. Al asignar a los usuarios diferentes canales de frecuencias, confinar las señales a límites de ancho de banda especificados, y restringir la potencia que se puede utilizar para modular las señales, se puede evitar la indeseada diafonía, es decir, las interferencias entre diferentes usuarios. Estas reglas, impuestas por agencias reguladoras de cada país, son necesarias porque cualquier aumento en la tasa de modulación amplía el ancho de banda de la señal de radio, lo cual aumenta la posibilidad de diafonía.

La principal ventaja de las ondas de radio con espectro ensanchado es que permiten el tratamiento de las señales para que se propaguen adecuadamente por el aire, a pesar de las interferencias electromagnéticas, eliminando virtualmente la diafonía. En la modulación por espectro ensanchado, se dispersa la potencia de una señal sobre una banda de frecuencias mayor. Esto resulta en una señal más robusta, que es menos susceptible a las interferencias de sistemas de radios similares, ya que éstos también están expandiendo sus señales, pero con diferentes secuencias de ensanchamiento.

El espectro ensanchado tiene dos modos de funcionamiento: Espectro Ensanchado por Saltos de Frecuencia (*Frequency Hopping Spread Spectrum*, FHSS) y Espectro Ensanchado de Secuencia Directa (*Direct Sequence Spread Spectrum*, DSSS). Los saltos de frecuencia ensanchan la señal “haciendo saltar” la señal de banda estrecha sobre toda la banda de radio en función del tiempo. La secuencia directa expande toda la señal a la vez sobre toda la banda de radio. Aunque la tecnología inalámbrica Bluetooth utiliza el modo FHSS, es interesante contrastarlo con el mecanismo de secuencia directa para apreciar sus ventajas.

**1.2.2 Espectro ensanchado por secuencia directa (DSSS)** En el ensanchamiento por secuencia directa, la energía de radio se dispersa por una porción de banda mayor de la necesaria para la transmisión de los datos. Esto se hace dividiendo cada bit de datos en múltiples sub-bits llamados chips, para crear una tasa de modulación mayor. Esta tasa se consigue multiplicando la señal

digital por una secuencia de ensanchamiento. Si la secuencia de ensanchamiento es de 10, por ejemplo, y se aplica a una señal que transporta datos a 300 Kbps, el ancho de banda resultante será 10 veces más amplio. El ensanchamiento de la señal depende del número de chips por cada bit de información.

Con la modulación de datos se amplía la portadora de radio a anchos de banda cada vez mayores a medida que se incrementa la velocidad de datos, esta tasa de expansión de 10 veces la velocidad de datos genera una portadora de radio 10 veces más amplia de lo que sería si solo hubiera datos. Utilizando el mismo código que el transmisor, el receptor puede correlacionar y devolver la señal a su forma original, mientras que otros receptores con diferentes códigos no lograrán recuperar la señal original.

Esta característica del espectro ensanchado hace posible construir y hacer funcionar múltiples redes en una misma ubicación. Al asignar a cada red una secuencia de ensanchamiento exclusiva, todas las transmisiones pueden compartir la misma banda de frecuencias y seguir siendo independientes unas de otras. Las transmisiones de una red aparecen ante la otra como ruido aleatorio y son filtradas porque las secuencias de ensanchamiento no coinciden.

Esta técnica de ensanchamiento podría dar como resultado una Relación Señal a Ruido (*Signal Noise Ratio*, SNR) más débil, ya que el proceso de ensanchamiento disminuye la potencia de la señal en cualquier frecuencia dada la dispersión de

energía sobre una banda mayor. Normalmente, una relación señal al ruido baja daría como resultado paquetes de datos dañados que necesitarían retransmitirse. Sin embargo, la ganancia de procesamiento del correlacionador en el receptor es el encargado de devolver la señal a su estado original, recuperando la pérdida de potencia cuando la señal se contrae a su ancho de banda original, aunque no la refuerza mucho más de lo que se habría recibido si la señal no se hubiera ensanchado. El número de chips está directamente relacionado con la inmunidad de la señal a las interferencias. En una zona con muchas interferencias de radio, los usuarios habrán de reducir la velocidad de transferencia de datos para limitar con éxito las interferencias.

**1.2.3 Espectro Ensanchado por Salto de Frecuencia (FHSS)** La tecnología inalámbrica Bluetooth utiliza la versión de Saltos de Frecuencia de Espectro Ensanchado (FHSS), que supone el salto del transmisor de una frecuencia a la siguiente con una velocidad de saltos específica de acuerdo con una secuencia de código pseudoaleatoria. El orden de las frecuencias seleccionadas por el transmisor se toma de un grupo predeterminado impuesto por la secuencia de código. Por ejemplo, el transmisor puede tener un patrón de salto que vaya de la tercera frecuencia a la duodécima, a la quinta, y así sucesivamente por todo el intervalo de frecuencias. El receptor no pierde de vista estos cambios de frecuencia. Como sólo el receptor deseado conoce el patrón de saltos del transmisor, solo ese receptor puede entender los datos transmitidos. El estándar

Bluetooth especifica una velocidad de 1.600 saltos por segundo entre 79 frecuencias.

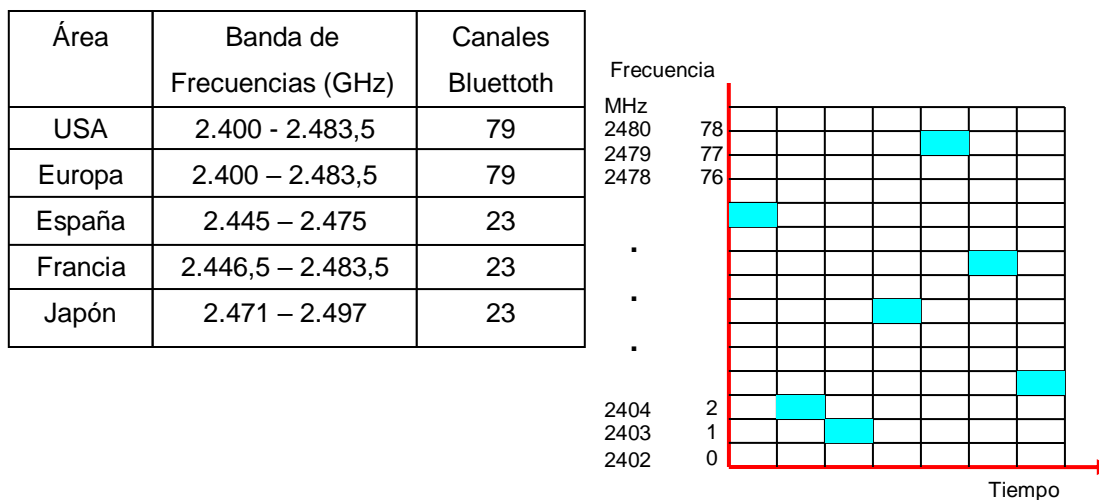


Figura 9. Saltos de frecuencia.

Todas las unidades Bluetooth participan en una pico-red, con todas las unidades compartiendo un canal común. Una pico-red puede soportar hasta ocho dispositivos interconectados, con un maestro y hasta siete esclavos. Esta relación continua mientras dura la conexión de la pico-red. Las unidades que participan en la pico-red están sincronizadas desde el punto de vista del tiempo y de los saltos en el mismo canal. Todas las unidades Bluetooth tienen un reloj de sistema interno que determina la temporización y los saltos que utiliza su transceptor. El reloj del maestro determina la sincronización y los saltos de frecuencia en una pico-red. Cuando se establece la pico-red, el reloj maestro se comunica con los esclavos. Cada esclavo agrega un desplazamiento a su reloj interno para sincronizarse con el reloj maestro. Como los relojes funcionan con independencia, se han de utilizar frecuentemente dichos desplazamientos.

Otros transmisores con saltos de frecuencia situados en las cercanías utilizarán diferentes patrones de saltos y velocidades de salto muchos más lentas que los dispositivos Bluetooth. Si los transmisores que no utilizan la tecnología inalámbrica Bluetooth intentaran incidentalmente utilizar la misma frecuencia en el mismo momento, el paquete de datos transmitido por uno o ambos dispositivos se verá corrompido debido a la colisión, por lo que se necesitará la retransmisión de los paquetes de datos afectados; se enviará otra vez un nuevo paquete de datos en el siguiente ciclo de saltos de cada transmisor.

Aunque la posibilidad de que dispositivos que utilicen la tecnología inalámbrica Bluetooth interfieran con los que no la utilizan, pero que comparten la misma banda de 2,4 GHz, es mínima, algunos fabricantes están preocupados, por lo que han pedido nuevas directrices que proporcionen más separación entre los protocolos rivales o dispositivos que trabajan en la banda de 2,4 GHz. El impacto de cualquier cambio se debe considerar tanto para teléfonos portátiles como para otros dispositivos como hornos microondas, altavoces inalámbricos y sistemas de seguridad, entre otros dispositivos que también comparten esta banda.

#### **1.2.4. Ventajas y desventajas de los dos tipos de espectro ensanchado**

Ventajas de DSSS:

- Permite mayores velocidades de datos (5,5 Mbps y 11 Mbps).

- Menor complejidad en comparación con los sistemas FHSS, ya que éstos siempre transmiten en un único canal.

#### Desventajas de DSSS:

- En un área sólo pueden funcionar cierta cantidad limitada de sistemas de forma simultánea.
- Necesita componentes más rápidos y caros que los sistemas FHSS equivalentes.
- Más consumo y requisitos que los sistemas FHSS.
- Más vulnerable a las interferencias de banda estrecha.

Los productos Bluetooth utilizan la técnica de transmisión denominada FHSS que está diseñada especialmente para garantizar que sus transmisiones sean seguras, sólidas y menos propensas a las interferencias.

#### Ventajas de FHSS:

- Alta tolerancia a interferencias
- Alta seguridad contra interceptación de señal

#### Desventajas de FHSS:

- Baja / Media velocidad
- Difícil de sincronizar a medida que la distancia entre dispositivos se incrementa. [4], [5], [6], [7].



## 2. METODOLOGÍA PARA LA SIMULACIÓN DE EQUIPOS DE TELECOMUNICACIONES

Se propone un esquema de acuerdo a las recomendaciones planteadas por la metodología secuencial aplicada a la simulación que en su orden se ubican de la siguiente manera: el qué, el cómo y la corrección de errores. Además los pasos corresponden al estudio de simulación de eventos discretos, acogiendo las características del análisis y diseño orientado a objetos. [8].

Tabla 4. Pasos de una simulación

<b>Pasos recomendados:</b>
- Formulación del problema y plan de estudios
- Recolección y procesamiento de datos
- Formulación de un modelo de simulación.
- Evaluación del modelo y los parámetros estimados.
- Programación.
- Validación del programa.
- Diseño de experimentos.
- Análisis de los datos obtenidos con la simulación.
- Documento e implementación de resultados

La figura 10, por su parte, muestra las vías de desarrollo a tomar entre uno y otro paso de la metodología. Se comienza con la formulación del problema inicialmente planteado por el interesado en el proyecto, buscando establecer qué se quiere, para qué se quiere y que se espera de la simulación. Una vez definido el problema se debe buscar un conocimiento más profundo y detallado sobre el mismo, para esto se desarrolla un plan de estudio que debe estar basado en criterios para evaluar la eficiencia en la consecución de la información en término del número de personas, costos y tiempo definido para cada aspecto del estudio.

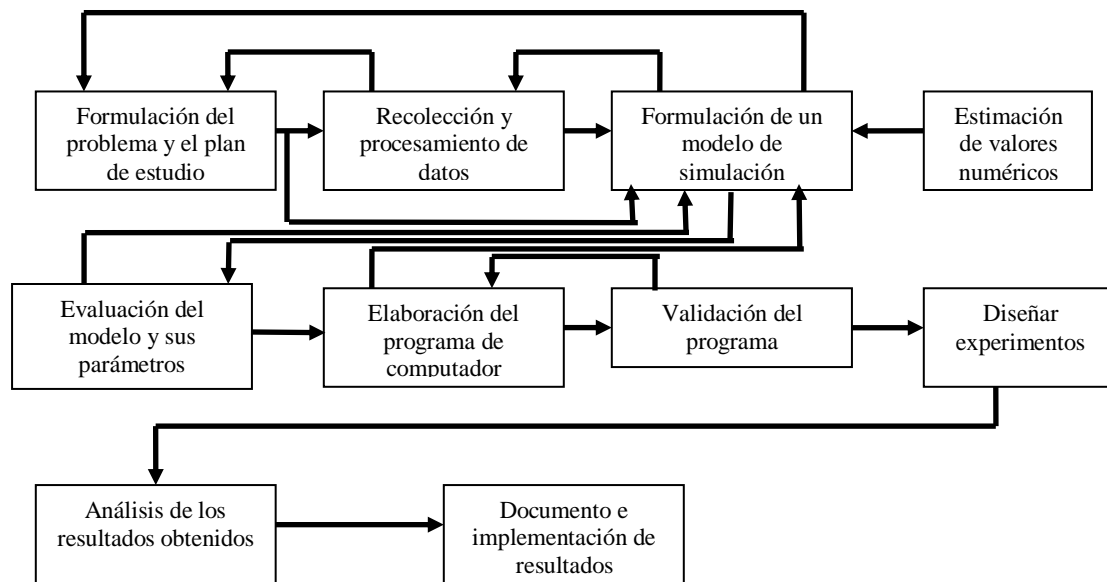


Figura 10. Vías de desarrollo para un trabajo de simulación.

Con la información recolectada se comienza a detallar las variables de entrada y de salida y los diferentes procesos que intervienen, donde finalmente se llega a un modelo que los representa.

Una vez planteado el modelo (Un modelo o paradigma es una representación de la naturaleza, para el cual se toman las propiedades más importantes y se desechan los aspectos que son considerados irrelevantes), se hace necesario realizar un estudio del comportamiento de cada variable de entrada y de sus efectos en la salida hasta obtener los valores numéricos necesarios para su representación. Para lograrlo se analiza el comportamiento de cada variable en su propio dominio.

Con los valores numéricos se procede a comprobar la validez del modelo mediante la realización de pruebas para hacer los ajustes necesarios a los valores estimados, o de ser necesario, replantear el modelo hasta obtener uno que satisfaga los requerimientos planteados.

Una vez obtenido el modelo más cercano a los intereses del proyecto se procede a desarrollar la simulación y se realizan los ajustes necesarios para obtener resultados satisfactorios.

## **2.1 FORMULACIÓN DEL PROBLEMA Y PLAN DE ESTUDIOS**

Cada estudio debe comenzar con una declaración clara de los objetivos que motivan la simulación. Los objetivos deben refinarse antes de proseguir con el desarrollo de la simulación. Los objetivos de la simulación deben planearse de

modo que aporten elementos para definir las variables de salida y la forma como van a presentarse y las variables o parámetros de entrada que afectarán esa salida. Una mala selección de un objetivo se reflejará en una selección incorrecta de los parámetros de entrada y un resultado insatisfactorio.

Para llegar a la correcta formulación de los objetivos de la simulación se recomienda seguir los siguientes pasos:

- Adquirir información sobre el funcionamiento del sistema.
- Identificar los fines de la simulación.

## **2.2 RECOLECCIÓN Y PROCESAMIENTO DE DATOS**

Los datos son tomados de la realidad o de experiencias previas, la información que se recopile debe ser descriptiva de forma que sea posible obtener un modelo modificable del sistema. Los objetivos fijados dan idea al diseñador sobre las variables de entrada y salida que debe manejar el modelo de simulación y así mismo se obtiene una visión clara de los datos que necesitará él mismo en su funcionamiento.

## 2.3 FORMULACIÓN DE UN MODELO DE SIMULACIÓN

Elaborar un modelo, considerando los aspectos de mayor interés, implica realizar una abstracción del sistema real a partir de un amplio conocimiento del mismo.

Abstracción es el principio de ignorar aquellos aspectos de un objeto que no sean relevantes a un propósito general en orden a concentrarse más fijamente en aquellos que si lo son.

La abstracción debe permitir reducir la complejidad de un sistema. Cuando se está abstrayendo de un sistema en general se debe conservar siempre la claridad del planteamiento sobre el que se está trabajando ya que se podría llegar a conceptos tan reducidos que pueden hacer perder la claridad y hasta la finalidad del planteamiento inicial aumentando la complejidad del sistema en lugar de reducirla.

El objetivo principal en la formulación de un modelo de simulación es adaptar a la realidad el resultado del análisis hecho al sistema; es decir, comenzar a tener en cuenta aquellas cosas propias de la implementación y del entorno.

El modelo generado debe caracterizarse por su realismo y su simplicidad. Estos dos atributos resultan, la mayoría de las veces, de naturaleza opuesta. Debe

buscarse un compromiso entre realismo y complejidad tal que el modelo generado sea lo suficientemente satisfactorio, o cercano a la realidad.

El empleo de las técnicas de simulación no requiere una alta formación matemática por parte del desarrollador, sin embargo es aconsejable la intervención de personas expertas en el fenómeno que se desea modelar; esto evitará que las conclusiones a las que se llegue mediante la simulación sean erróneas.

## **2.4 EVALUACIÓN DEL MODELO Y LOS PARÁMETROS ESTIMADOS**

Son pruebas de ensayo y error que buscan ajustar los valores de los límites y los rangos considerados por las variables de los diferentes procesos (objetos), teniendo en cuenta los atributos de cada objeto y las relaciones funcionales respectivas de forma que los resultados obtenidos sean lo más cercano de la realidad.

Es necesario ser cuidadosos al evaluar el modelo, de manera que se puedan detectar los errores del desarrollo del mismo en esta etapa y evitar errores que obliguen a regresar a etapas anteriores a ésta, incrementando el tiempo y el costo del desarrollo de la solución.

Los factores que se deben tener en cuenta para realizar la evaluación son:

- Verificación de la correcta descripción de las variables. Se deben examinar las formas de almacenamiento y direccionamiento de las variables, tanto de entrada y salida como las intermedias. Evaluar, si es posible, que se estén empleando variables redundantes o que sean necesarias más variables de las utilizadas.
- Examinar la complejidad resultante para cada objeto (proceso) en que se ha dividido el sistema, considerando el número de atributos y servicios que posee, las variables que maneja y la interacción entre las mismas. Tener en cuenta la dependencia temporal y si su descripción es la adecuada o si se está subutilizando (podrán ampliarse sus atributos y servicios o dividirse en objetos). Considerar la posibilidad de que la población de datos dispuesta para estimar los parámetros no sea significativa frente a una expansión futura del modelo. Observar si las limitaciones impuestas al modelo pueden ser superadas en el futuro.
- Evaluar si el modelo generado satisface los objetivos planteados; además se deben considerar si se han estudiado desarrollos alternativos que puedan resultar de menor complejidad con igual o mejores resultados.
- Evaluar si con la complejidad del proyecto, de manera global, resulta procedente continuar.

- Por último, se debe evaluar en compañía de las personas interesadas en el desarrollo del proyecto de simulación, si el porcentaje en el cual se está cumpliendo con los objetivos propuestos inicialmente es tal que se puede continuar con él o es necesario el replanteamiento completo del modelo.

En general, la actividad de validación es algo que debe hacerse a lo largo de la realización del proyecto de simulación. Es imperativo para el modelador involucrar gente que esté íntimamente familiarizada con el sistema a simular.

## **2.5 PROGRAMACIÓN**

Una vez formulado y validado el modelo de simulación, se procede a programar basados en un software de computador existente que permita realizar la simulación del sistema. El programa debe cumplir con condiciones de operación, ampliación y mantenimiento impuestas por el sistema, por las personas interesadas en el proyecto y por la necesidad de modificaciones que puedan presentarse en el futuro.

El programa puede obtenerse de una de dos formas que son: El empleo de programas de computador (paquetes) diseñados para trabajo en simulación propiamente dicha o el uso de lenguajes de propósito general.



Los lenguajes de simulación proporcionan automáticamente muchas de las características necesarias en la programación de un modelo de simulación, favoreciendo la disminución del tiempo de programación, el cual a menudo resulta ser significativo. Proporcionan elementos y aspectos de interacción y comunicación entre bloques de forma natural y segura, favoreciendo la integración del programa.

Comúnmente los lenguajes de simulación proporcionan almacenamiento dinámico en memoria durante la ejecución. También una mejor detección de errores porque muchos tipos de estos errores potenciales han sido identificados y son verificados automáticamente.

A menudo los lenguajes de simulación proporcionan un ambiente interactivo en el cual el proceso de simulación puede ser interrumpido para permitir diferentes funcionalidades o fines tales como: realizar una pregunta de parte del usuario o para realizar un reporte. Con los reportes el usuario puede definir cual debe ser la modificación a realizar o lo que debe ejecutarse la próxima vez. Esta cualidad es muy importante durante la construcción de un modelo o durante el estudio de validación.

Los lenguajes de propósito general por su parte proporcionan ventajas tales como que la mayor parte de los programadores ya conocen y manejan un lenguaje de propósito general, lo que no es común cuando se trata de un lenguaje de simulación.

Un programa escrito utilizando un lenguaje de propósito general puede requerir un tiempo de menor ejecución que el correspondiente programa escrito en lenguaje de simulación. Esto se debe a que un lenguaje de simulación está diseñado para modelar un gran cantidad de sistemas con un determinado número de bloques de construcción mientras que un programa escrito en lenguaje C, por ejemplo, puede ser sencillo para una aplicación en particular. Un lenguaje de propósito general permite una mayor flexibilidad de programación que ciertos lenguajes de simulación.

Existe una gran variedad de lenguajes y paquetes destinados para realizar programas de simulación, así como también un programador puede contar con una amplia gama de lenguajes para programación de propósito general. Para decidir la elección de un lenguaje de programación, el diseñador deberá tener en cuenta entre otras cosas las siguientes consideraciones:

Periodo de tiempo disponible para realizar el programa. Si se cuenta con un periodo de tiempo restringido para la realización del programa, el diseñador debería optar por elegir un lenguaje de programación que le sea familiar y sobre el cual tenga dominio o escoger un lenguaje sobre el cual se hayan realizado anteriormente aplicaciones similares y que pueda tomarse como base para implementar su aplicación particular. Si el periodo de tiempo para realizar el programa es relativamente prolongado, el diseñador podrá empezar por conocer el lenguaje y los paquetes que le resulten accesibles para elegir finalmente aquel que se ajuste de la mejor manera a sus necesidades particulares.

## 2.6 VALIDACIÓN DEL PROGRAMA

La validación del programa consiste en la realización de una serie de pruebas encaminadas a comprobar la correcta ejecución del mismo, antes que comprobar la simulación propiamente dicha. Esto debido a que el proceso de validación de la simulación se hace posteriormente a la terminación del programa.

Las primeras pruebas están destinadas a comprobar la correcta ejecución de cada objeto como una unidad absoluta e independiente. Posteriormente se comprueba que el desempeño sea el esperado al realizar la integración de los módulos, es decir se evalúa la correcta arquitectura del programa.

Es necesario validar los requerimientos dados al iniciar la codificación para constatar que el programa terminado satisfaga las condiciones del modelo de simulación planteado.

Finalmente, se ejecuta sobre el sistema global una serie de pruebas para determinar su capacidad de recuperación frente a posibles errores introducidos por el usuario, la seguridad en el manejo de los datos necesarios y su rendimiento.

## **2.7 DISEÑO DE EXPERIMENTOS**

El diseñador debe tomar las condiciones más importantes que debe cumplir el sistema desarrollado para realizar experimentos que lleven a observar el comportamiento del mismo frente a tales condiciones, los resultados obtenidos de la ejecución de la simulación y el número de ejecuciones necesarias para asegurar una completa observación.

Los experimentos deben ser orientados a comprobar de manera detallada la ejecución del sistema y deben ser diseñados de tal manera que los casos críticos estén incluidos en la prueba y que ésta se haga eficiente en términos de tiempo, recursos y seguridad.

## **2.8 ANÁLISIS DE LOS DATOS OBTENIDOS CON LA SIMULACIÓN**

En este punto se deben tener en consideración los objetivos que fueron planteados para el trabajo de simulación, se hace interpretación de los mismos de forma que sea posible sacarle provecho a la simulación.

Quien hace el análisis puede estudiar y experimentar con las posibles interacciones que ocurren al interior del sistema y que generalmente resultan

altamente complejas. Se puede estudiar el efecto de ciertos cambios en la información suministrada como entrada al sistema, en la organización misma entre las partes del sistema y en las condiciones propias del entorno que rodea al sistema. Lo anterior conduce a un mejor entendimiento del comportamiento y evolución del sistema a través de diferentes situaciones que resulten de interés para el diseñador.

La interpretación de los resultados obtenidos con la simulación deben buscar elementos que permitan concretar el logro de los objetivos de la simulación y así poder adquirir un mayor conocimiento del sistema, de su comportamiento, de sus limitaciones, de sus potencialidades y de aquellas condiciones que deban modificarse para probarlo y mejorarlo antes de realizar la implementación en la realidad. Se puede experimentar con situaciones nuevas acerca de las cuales no se tiene mucha o ninguna información, de esta manera se estaría preparando al sistema para posibles eventualidades.

## **2.9 DOCUMENTO E IMPLEMENTACIÓN DE RESULTADOS**

Esta etapa consiste en la elaboración del documento formal de presentación del trabajo realizado, es preciso mencionar que la tarea principal que el diseñador debe hacer aquí es especificar de manera completa y clara el desarrollo de todas y cada una de las etapas que involucran el desarrollo de un trabajo de simulación.

Para la elaboración del documento y la implementación de los resultados se sugiere al diseñador plantearse los objetivos siguientes:

- Comprobar que los motivos de la realización del trabajo sean alcanzados al concluir el trabajo por el diseñador.
- Presentar la información sobre el sistema a simular que resulte necesaria para un conocimiento completo del mismo.
- Indicar y justificar todas las condiciones de operación que el diseñador considere, debe cumplir con el modelo de simulación por él planteado.
- Lograr que la presentación de análisis del sistema, el diseño y la implementación, incluyendo la elaboración del programa de computador resulte clara y precisa, siendo fácilmente entendible y modificable, si es preciso, por otro diseñador. Para lograr este objetivo es recomendable que el diseñador deje documentos, manuales, condiciones de operación y la mayor cantidad de documentación posible.

### **3. MODELO CONCEPTUAL DE BLUETOOTH**

Antes de efectuar cualquier paso en el proceso de diseño e implementación de cualquier tipo de sistema, es conveniente desde el punto de vista del ingeniero o desarrollador realizar un análisis del ambiente real en cuanto al propósito del objeto de desarrollo para definir claramente los métodos y variables implicadas en el sistema. En consecuencia a lo anterior, este capítulo plantea un modelo que contempla los conceptos involucrados en los niveles de enlace de datos y físico de la tecnología Bluetooth de manera que permite tener una referencia previa del sistema a construir dentro del marco de la simulación. Considerando el hecho de que se sigue una metodología para simulación de sistemas de telecomunicaciones, se hace énfasis en la formulación del modelo de simulación propuesto en ella, sin abordarla en su totalidad por el hecho de resultar en algunos aspectos redundante, siendo esto suficiente antes de simular, dado que reduce el tiempo de diseño y posibles fallas en depuración y funcionamiento.

### 3.1 CONTENIDO TEÓRICO

Para lograr una adecuada herramienta final de simulación, es esencial trabajar con información precisa, que para el caso se tendrá documentación desarrollada por el Grupo de Interés Especial de Bluetooth (Special Interest Group, SIG) bajo la normativa hecha por el IEEE sobre esta tecnología que en definitiva es la base conceptual para el modelado del sistema.

**3.1.1 Extracto de la norma IEEE 802.15.1 WPAN tecnología Bluetooth** Las especificaciones de Bluetooth definen los requisitos para comunicación de voz y datos sobre un canal de radio con una capacidad máxima de 1Mbps. Bluetooth transmite con bajo nivel de potencia (1mW) y por lo tanto es utilizado en transmisiones de corto alcance, menos de 10 metros. El esquema de la modulación utilizado por Bluetooth es GMSK. Los saltos de frecuencia se emplean para evitar la interferencia con otros dispositivos que transmiten en la misma banda. Aunque las transmisiones de Bluetooth colisionarán de vez en cuando con la de otro dispositivo, esto se puede tolerar y recuperar la información con esquemas de codificación apropiados. El tiempo de la transmisión se divide en intervalos de tiempo de 625 $\mu$ s, con una nueva frecuencia de salto utilizada para cada intervalo. Por otro lado, el ancho de banda empleado por la tecnología es mayor a 79MHz para simular el modelo exacto de los efectos de salto de frecuencia.



**3.1.2 Definición de canal** Como se comentó en el primer capítulo, Bluetooth utiliza un sistema FH / TDD (Salto en Frecuencia / Duplexación por División de Tiempo), en el que el canal queda dividido en intervalos de 625  $\mu$ s, donde cada salto de frecuencia se da al finalizar un intervalo de tiempo. Esto da lugar a 1600 saltos por segundo, en cada salto un paquete de datos ocupa un intervalo para la emisión y otro para la recepción y pueden ser utilizados alternadamente, dando lugar a un esquema del tipo Duplexación por División de Tiempo (*Time División Duplex, TDD*). Ver figura 11.

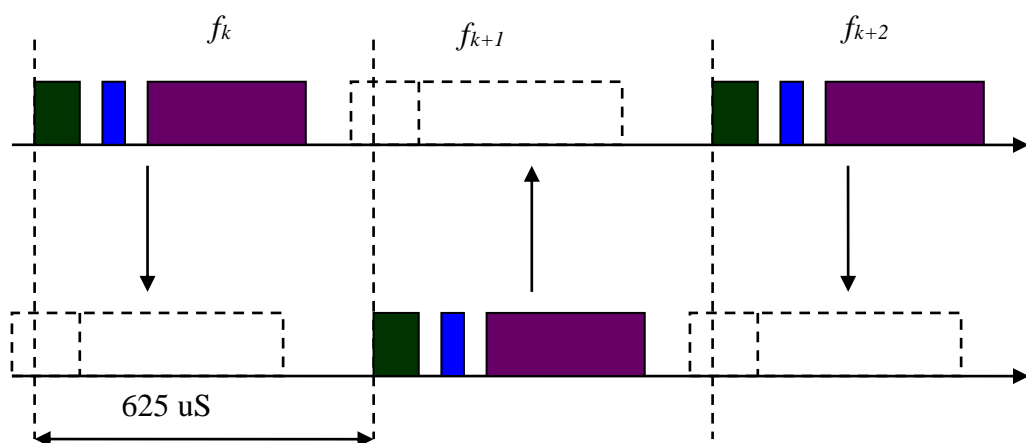


Figura 11. Intervalos de tiempo.

Dos o más unidades Bluetooth pueden compartir el mismo canal dentro de una pico-red la cual es una red de interiores o instalaciones (indoor), donde una unidad actúa como maestra y las otras como esclavas, enviando y recibiendo señales hacia el maestro. Los saltos de frecuencia están determinados por la secuencia de ensanchamiento utilizada en la codificación. En Bluetooth, la secuencia de ensanchamiento se determina por la identidad de la unidad maestra de la pico-red (un código único para cada equipo) y por su frecuencia de reloj.

En países donde la banda está abierta a 80 canales o más, espaciados todos ellos a 1 MHz, se han definido 79 saltos de portadora, y en aquellos donde la banda es más estrecha se han definido 23 saltos.

**3.1.3 Definición de paquete** La información que se intercambia entre dos unidades Bluetooth se realiza mediante un conjunto de intervalos de tiempo que forman un paquete de datos. Cada paquete comienza con un código de acceso de 72 bits, que se deriva del código de identidad de la unidad maestra, seguido de un paquete de datos con un encabezado de 54 bits. Éste contiene importante información como tres bits de dirección de acceso, tipo de paquete, bits de control de flujo, bits para la Solicitud de Retransmisión Automática (*Automatic Request Query*, ARQ) y chequeo de errores de campos de encabezado. Finalmente, el paquete que contiene la información, que sigue al del encabezado, tiene una longitud de 0 a 2745 bits. En cualquier caso, cada paquete que se intercambia en el canal está precedido por el código de acceso.

Los receptores de la pico-red comparan las señales que reciben con el código de acceso, si éstas no coinciden, el paquete recibido no es considerado como válido en el canal y el resto de su contenido es ignorado.

Cabe aclarar que la especificación de Bluetooth, y de enlaces de comunicación en general, comprenden mucho más que la capa física. En particular, los componentes del control lógico presentan esquemas de reconocimiento,

protocolos, y las capas más altas (2 y 3) son una parte importante en el desafío de diseño. Tal funcionalidad es difícil de planear como un diagrama de bloques, pero es fácil de planear como Máquina de Estados Finitos (*Finite State Machine*, FSM), no obstante este conjunto de tareas relacionadas con el diseño del programa de simulación a ese nivel no hace parte del objetivo central de este trabajo de grado quedando como una posible propuesta de evolución al presente proyecto.

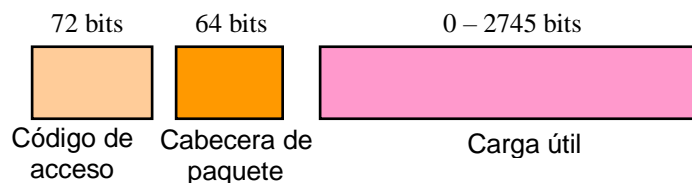


Figura 12. Estructura de un paquete.

**3.1.4 Definición de enlace físico** En la especificación Bluetooth se han definido dos tipos de enlace que permitan soportar incluso aplicaciones multimedia:

- Enlace Síncrono Orientado a la Conexión (SCO, *Synchronous Connection Oriented*).
- Enlace Asíncrono no Orientado a la Conexión (ACL, *Asynchronous ConnectionLess*).

Los enlaces SCO soportan conexiones asimétricas, punto a punto, utilizadas normalmente en conexiones de voz. Dentro del canal de comunicaciones establecido se les reservan dos intervalos de tiempo consecutivos (envío y

retorno) en intervalos fijos. Los enlaces ACL soportan conmutaciones punto a punto simétricas o asimétricas, típicamente empleadas en la transmisión de datos. Con relación a cada tipo de enlace se han definido un conjunto de paquetes.

Para los enlaces SCO, existen tres tipos de intervalos de tiempo simples, cada uno con una velocidad asociada de 64 Kbps que corresponde a la tasa de información estandarizada para transmisión de voz. La transmisión de voz se realiza sin ningún mecanismo de protección, pero si el intervalo correspondiente a cada una de las señales en el enlace SCO disminuye, se puede seleccionar una velocidad de corrección de envío de 1/3 o 2/3 garantizando un determinado nivel de integridad de la información.

Para los enlaces ACL, se han definido los intervalos 1,3 y 5. Cualquiera de los datos puede ser enviado protegido o sin proteger con una velocidad de corrección de 2/3. La máxima velocidad de envío es de 721 Kbps en una dirección y de 57,6 Kbps en la otra.

En Bluetooth, la transmisión de voz se realiza solamente cada sexto intervalo. Este período, igual a 3,75ms, se denota como  $T_{SCO}$ . El retorno (trayectoria esclavo a maestro) transmite en el siguiente intervalo según lo mostrado en la figura 13. Hasta tres llamadas de voz simultáneas pueden ser soportadas de esta manera. La figura 13 muestra las ranuras o intervalos de transmisión para cada enlace maestro SCO (T1, T2, y T3) y los intervalos de la trayectoria de regreso para los esclavos (R1, R2, y R3).

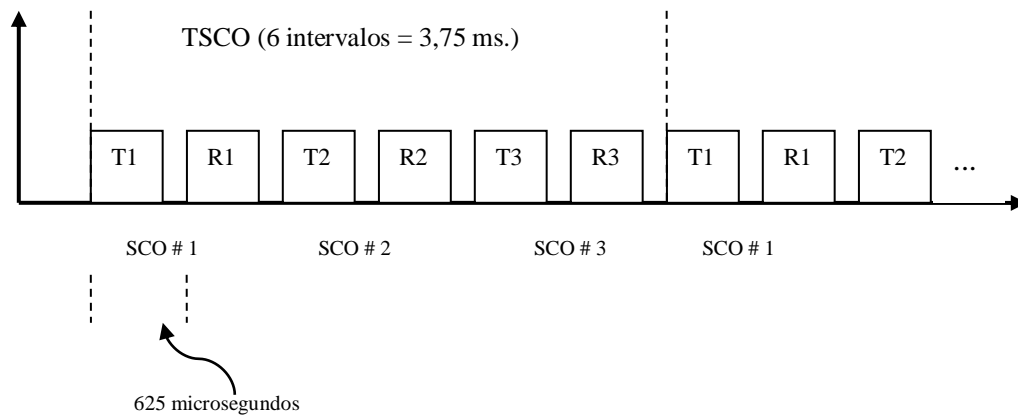


Figura 13. Diagrama de tres llamadas de voz simultáneas.

**3.1.5 Transmisión de datos** La transmisión de datos utiliza paquetes ACL. Una diferencia importante respecto a la transmisión de voz, es que la transmisión de los datos tiene que ser perfecta (es decir, no se permiten errores). En términos generales, respecto a los sistemas de comunicaciones cuando se descubre un error en una trama, el dato errado tiene que ser retransmitido. Hay varias formas diferentes de lograr esto en el diseño de sistemas de comunicación. En Bluetooth, los paquetes son enviados repetidamente por el transmisor hasta que reciba un reconocimiento por parte del receptor, que notifica que el paquete fue recibido correctamente. El receptor prueba la Codificación de Redundancia Cíclica (*Cyclic Redundancy Code*, CRC) del paquete entrante para determinar que la recepción fue correcta. En la transmisión de datos es necesario estimar el rendimiento de procesamiento, en unidades de bits por segundo, en lugar de la BER. El rendimiento de procesamiento se reduce si es necesario reenviar paquetes.

El receptor envía reconocimientos de retorno al transmisor poniendo el bit de estado del ARQN (Acknowledge Request N) en la información del encabezado en el camino de retorno del paquete. El valor de este bit señala al dispositivo transmisor si el último paquete fue recibido correctamente. ARQN=1 significa una transmisión exitosa, ARQN=0 significa transmisión no exitosa.

Si los datos necesitan ser comunicados solamente del maestro al esclavo, el esclavo envía de regreso un paquete NULO simulado sin carga útil en el siguiente intervalo de tiempo. La figura 14 muestra, como ejemplo, el paquete DM1 transmitiéndose en el primer intervalo de tiempo, y el esclavo que contesta con un paquete NULO que contiene el reconocimiento en el intervalo de tiempo que sigue inmediatamente. El maestro transmite entonces de nuevo en el próximo intervalo de tiempo.

En caso de necesitar un modelo de comunicación bidireccional, se necesitaría diseñar dos dispositivos idénticos, cada uno con un transmisor y receptor.

En los estados de la especificación Bluetooth el esquema del reconocimiento es como sigue: Con un esquema de solicitud de respuesta automática, DM1 (paquetes de datos cortos), DH (paquetes de datos largos), y el campo de datos de paquetes DV (paquetes de voz y datos), son transmitidos y retransmitidos hasta que el reconocimiento de una recepción exitosa es retornado por el destinatario. La información de reconocimiento se incluye en el encabezado del paquete de retorno.

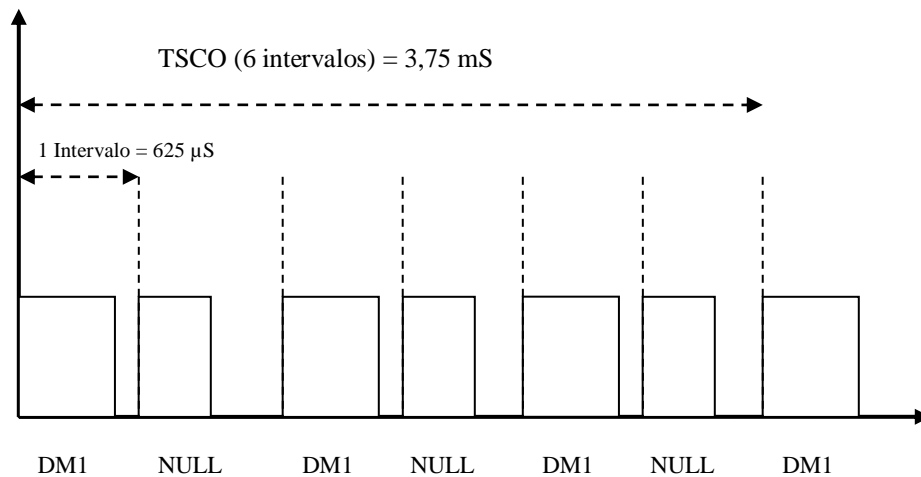


Figura 14. Registro de tiempo de paquetes DM1 y NULL.

**3.1.6 Reconocimiento de los datos en el transmisor** El primer paso en el diseño del algoritmo correspondiente es identificar los diferentes estados en los que podría estar el dispositivo. Al transmitir paquetes ACL, el transmisor puede estar en cualquiera de estos dos estados (bajo el concepto de máquina de estados finitos): "Transmitiendo un nuevo paquete" o "Re-transmitiendo un paquete", etiquetando la transmisión de paquete en blanco ("Transmit\_blank\_Packet") y haciendo una Retransmisión de paquete ("Re\_Transmit\_Packet") respectivamente. El subestado "paquetes ACL" inicia cuando el paquete es del tipo DMI, que es chequeado por el "paquete de tipo = 4, [Packet\_Type == 4]". Al iniciar en el estado "ACL\_Packets", la primera transición del dispositivo es al subestado por defecto que es "Transmit\_blank\_Packet". Los estados de acción "Enable\_Audio=0" y "Get\_blank\_Packet=1" envían señales por fuera de los caracteres de estado para desactivar el audio y generar un nuevo paquete de bits

respectivamente. El transmisor se queda en este estado hasta el próximo intervalo de tiempo a transmitir. En ese momento, él "despierta" y mira el estado del bit ARQN en el paquete que se retornó desde dispositivo receptor en la ranura anterior. Si el transmisor está en el estado "Transmit blank Packet" y el retorno ARQN =1, se queda en ese estado y transmite otro nuevo paquete. Esto se lleva a cabo en un diagrama de flujo de estados poniendo las condiciones "[ARQN == 0]" o "[ARQN == 1]" en las transiciones de los estados salientes. Si el retorno ARQN = 0 las transiciones pasan al estado "Re\_Transmit\_Packet". Cuando el transmisor está en estado "Re\_Transmit\_Packet" y el retorno es ARQN =1, las transmisiones regresan al estado "Transmit blank Packet", y si ocurre lo contrario (ARQN = 0) continua en la condición "Re\_Transmit\_Packet", (figura 15).

### **3.2 MODELOS CONCEPTUALES**

Para la formulación del modelo de simulación de la tecnología Bluetooth, lo que se define es un modelo previo basado en conceptos que permita tener una representación abstracta de menor complejidad que el modelo de simulación como tal. Dentro del modelo (el conceptual) se muestran tanto conceptos base como atributos de los mismos y las asociaciones que puedan existir entre conceptos.



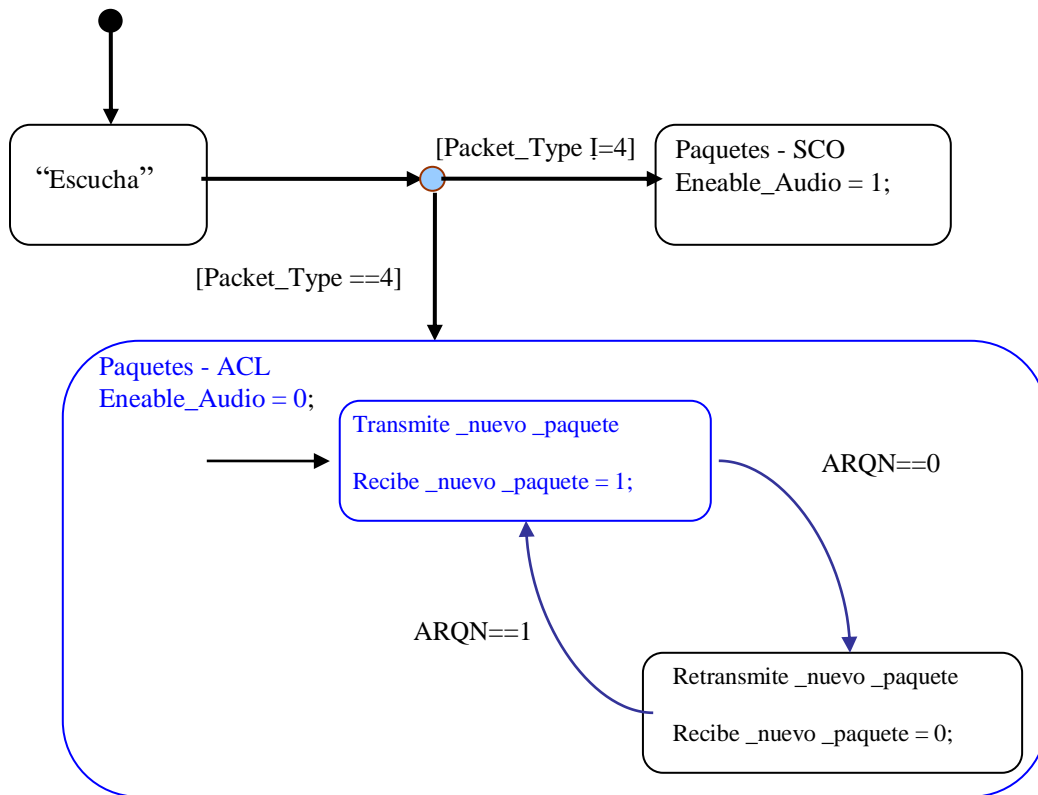


Figura 15. Diagrama de estados.

La palabra **conceptual** se refiere a creencias teóricas. La palabra modelo se refiere a una representación simplificada de la realidad. Un **Modelo conceptual** es por lo tanto, una representación de las creencias teóricas en cuanto al proyecto a desarrollar y obedece a un diagrama mediante el cual se ilustra una serie de relaciones entre ciertos factores que se cree impactan o conducen a una condición de interés.

Los modelos presentan un marco conceptual donde reflejan las teorías, plasman propiedades y establecen los principios del diseño de los sistemas. Su

importancia radica en que permiten identificar, organizar y realizar razonamientos sobre los componentes y comportamientos del sistema, son la guía para el proceso de diseño del software (en este caso el de simulación) y pueden emplearse posteriormente como una referencia para evaluar un diseño particular.

Dentro de los modelos conceptuales, se puede optar por métodos formales o notaciones ampliamente utilizadas y conocidas para representar los conceptos más relevantes.

En el Modelo Conceptual se tiene una representación de conceptos del mundo real, no de componentes software.

El objetivo de la creación de un Modelo Conceptual es facilitar la comprensión del problema. Por tanto, a la hora de incluir conceptos en el modelo, es mejor crear un modelo con muchos conceptos que quedarse corto y olvidar algún concepto importante, categorizándolos y determinando características, llamadas también atributos, y que dentro del ambiente de cualquier programa conforman las variables, parámetros específicos que caracterizan un sistema de manera particular y asociaciones entre los conceptos que indican relaciones entre ellos y muestran sus interacciones.

**3.2.1 Identificación de conceptos dentro de la tecnología Bluetooth** Para identificar el conjunto de conceptos se analizaron las especificaciones, los requisitos del sistema y el objetivo del proyecto.

Desde esa perspectiva se establecieron posibles categorías para el sistema, las cuales encierran un conjunto de conceptos mediante los que se describe el funcionamiento de la tecnología Bluetooth dentro de la norma IEEE 802.015.1 y que en adelante se llamará **dominio de conceptos** de Bluetooth. A continuación se listan los conceptos relevantes que son fundamentales para la obtención del modelo de simulación bajo Matlab®:

### **Categorías de Conceptos**

Señal

Modulación

Demodulación

Banda

Espectro

Canal

Secuencia

Control de errores

Intervalo de tiempo

Almacenador o buffer

Ensamble

Desensamble

Receptor

Trasmisor

Todos estos son nombres o sustantivos que están dentro del dominio o universo de la tecnología Bluetooth, también se encuentran dentro del vocabulario específico del objeto de desarrollo.

**3.2.2 Identificación de asociaciones del sistema Bluetooth** Una asociación es una relación entre conceptos que indica una conexión con sentido y que es de interés en el conjunto de los posibles procesos o métodos implicados en el funcionamiento de un sistema.

Dentro de las capas inferiores del sistema Bluetooth se incluyen las asociaciones siguientes:

- La señal tipo N es un elemento físico de entrada al transmisor.
- La unidad B codifica y ejecuta el método de corrección de errores sobre la información lógica.
- La unidad o módulo C ensambla paquetes de 625 bits actuando sobre la señal codificada (recibe la salida de la unidad B).
- La unidad o módulo D marca el salto de frecuencia codificada es decir gestiona la salida de C estableciendo el cambio de frecuencias.
- El módulo o unidad E efectúa la modulación de banda base sobre la salida de D adaptándola a la unidad F.

- La unidad F efectúa el acople al elemento físico de transmisión.
- La unidad G realiza el cambio o demodulación de señal física del elemento que soporta la transmisión y que se denomina canal de transmisión.
- La unidad C desensambla los paquetes adaptando la información al elemento lógico de corrección de errores.
- La unidad B aplica el método de verificación de errores después de capturar la información desensamblada.
- Las capas superiores externas interpretan la señal obtenida en el extremo final del receptor.

**3.2.3 Identificación de Atributos** Es necesario incorporar al Modelo Conceptual los atributos necesarios para satisfacer las necesidades de información de los métodos o procedimientos del sistema, en este caso de un transmisor y un receptor Bluetooth.

Los atributos deben tomar valor en tipos simples (número, texto, etc.), pues los tipos complejos deberían ser modelados como conceptos y ser relacionados mediante asociaciones, los atributos o variables identificables en el sistema son:

Concepto: señal

Atributos: binaria 0 o 1, codificada / decodificada, tipo voz, modulada tipo Delta con Pendiente de Variable Continua (*Continuously Variable Slope Delta*)

*Modulation, CVSD*) (opcional), modulada con Cambio de Frecuencia Mínima Gaussiana (*Gaussian Minimum Shift Keying, GMSK*), radiada en Canal con Ruido Blanco Gaussiano Aditivo (*Additive White Gaussian Noise, AWGN*), RF (Radiofrecuencia) en la banda ISM, Modulada FM.

Concepto: Control de errores  
Atributos: Codificar, decodificar

Concepto: Ensamble  
Atributos: Almacenamiento, paquetes, tamaño, velocidad, trama.

Concepto: Modulación  
Atributos: Salida GMSK, salida CVSD (opcional)

Concepto: Demodulación  
Atributos: FM , entrada CVSD (opcional)

Concepto: Canal  
Atributos: SCO, ACL, AWGN

Concepto: Secuencia  
Atributos: Salto aleatorio.

Concepto: Espectro  
Atributos: Ensanchado

Concepto: Banda  
Atributos: ISM, Ancho, Base

Concepto: Intervalo de tiempo  
Atributos: 625  $\mu$ s

Concepto: Transmisor

Atributos: Emisión

Concepto: Receptor

Atributos: Recepción

### 3.2.4 Fundamentación del modelo conceptual del sistema Bluetooth

A continuación se detalla la elaboración en notación formal y genérica del modelo conceptual del sistema transmisor / receptor Bluetooth, que es el aspecto central del presente capítulo.

- **Diagramas de conceptos (representaciones formales)** A continuación se toman los conceptos y sus atributos para crear una abstracción completa mediante la cual se describe el funcionamiento de un enlace completo entre un transmisor y un receptor Bluetooth. Para ello, se abordan los conceptos de manera modular e individual, buscando efectuar un análisis más efectivo que permita visualizar los atributos relacionados con cada uno, y posteriormente abordar las asociaciones entre los módulos.

#### – Señal

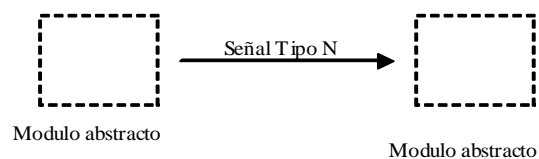


Figura 16. Representación señal.

El universo de atributos del concepto Señal incluye, para el caso,  $N = \{D, SC, DC, V, MC, RA, MG, RFISM, DFM\}$ , donde:

Tabla 5. Atributos del concepto señal

Sigla	Descripción
D	Datos
SC / DC	Codificada / Decodificada
V	Voz
MC (opcional)	Modulada CVSD
MG	Modulada GFSK
RA	Radiada AWGN
RFISM	En RF Banda ISM
DFM	Demodulada FM

- **Control de errores**

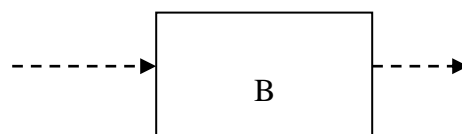


Figura 17. Representación control errores.



El universo de atributos del concepto B incluye los atributos  $CD = \{C (15,10), D (15,10)\}$  los cuales manifiestan sendas relaciones numéricas dependientes de un tratamiento aritmético. Especificando:

Tabla 6. Atributos del concepto control de errores

Sigla	Descripción
C (15,10)	Codificador (15,10)
D(15,10)	Decodificador(15,10)

- **Ensamble de paquetes**

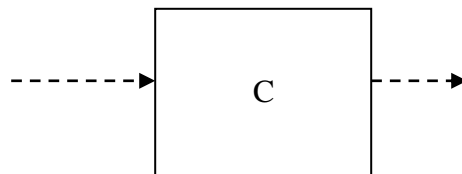


Figura 18. Representación ensamble.

El universo de atributos del concepto C incluye  $EP = \{AL, PQ, TM, VEL, TR, DES\}$ , siendo:

Tabla 7. Atributos del concepto ensamble

Sigla	Descripción
AL	Almacenamiento
PQ	Paquete
TM	Tamaño
VEL	Velocidad
TR	Trama
DES	Desensamble de Paquetes

- **Modulación**

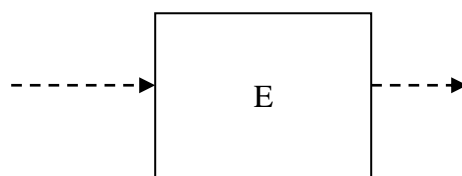


Figura 19. Representación modulación.

El universo de atributos del concepto E incluye  $M = \{SG, SCV \text{ (opcional)}\}$ , siendo:

Tabla 8. Atributos del concepto modulación

Sigla	Descripción
SG	Salida GMSK
SCV	Salida CVSD (opcional)

- **Demodulación**

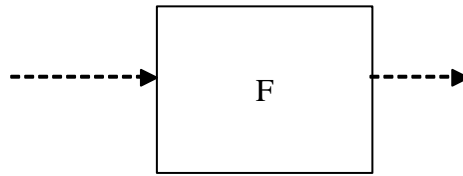


Figura 20. Representación demodulación.

El universo de atributos del concepto F incluye  $D = \{EG, ECV \text{ (opcional)}\}$ , siendo:

Tabla 9. Atributos del concepto demodulación

Sigla	Descripción
EG	Entrada a G en Frecuencia Modulada
ECV	Entrada CVSD (opcional)

- **Canal**

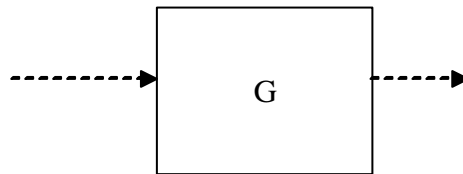


Figura 21. Representación canal.

El universo de atributos del concepto G incluye  $C = \{SCO, ACL, AWGN\}$ , siendo:

Tabla 10. Atributos del concepto canal

Sigla	Descripción
SCO	Canal Orientado a Conexión (opcional)
ACL	Canal no Orientado a Conexión (opcional)
AWGN	Canal de ruido Blanco

- **Secuencia**

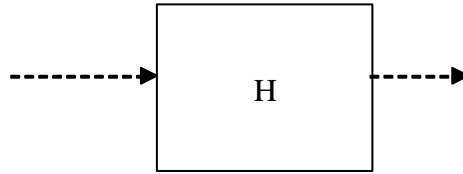


Figura 22. Representación secuencia.

El universo de atributos del concepto H incluye  $FH = \{PA, SF\}$ , siendo:

Tabla 11. Atributos del concepto secuencia

Sigla	Descripción
PA	Pseudo Aleatoriedad
SF	Salto en Frecuencia

- **Espectro**

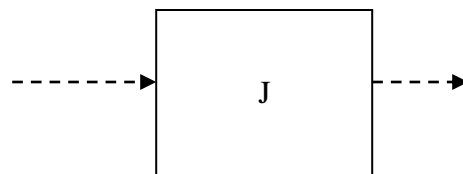


Figura 23. Representación espectro.

El universo de atributos del concepto J incluye  $ESP = \{D\}$ , siendo:

Tabla 12. Atributos del concepto espectro

Sigla	Descripción
D	Disperso o Ensanchado

- **Banda**

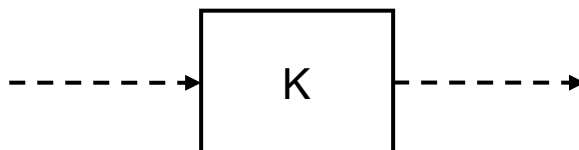


Figura 24. Representación banda.

El universo de atributos del concepto K incluye  $BAN = \{ISM, W, BAS\}$ , siendo:

Tabla 13. Atributos del concepto banda

Sigla	Descripción
ISM	Banda Industrial, Científica y Médica de 2,4 GHz
W	Ancho
BAS	Base

- **Intervalo de tiempo**

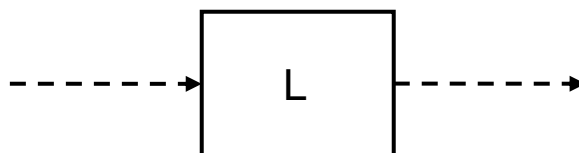


Figura 25. Representación del intervalo.

Cuyo universo de atributos de L incluye  $ESP = \{625\}$ , donde:

Tabla 14. Atributos del concepto intervalo

Sigla o valor	Descripción
625	625 microsegundos

**- Transmisor**

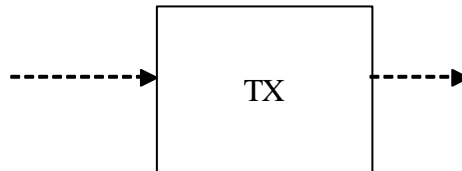


Figura 26. Representación transmisor.

Este concepto (TX) incluye en su universo de atributos el conjunto  $TR = \{EM\}$ , siendo que:

Tabla 15. Atributos del concepto transmisor

Sigla	Descripción
EM	Emisión

**- Receptor**

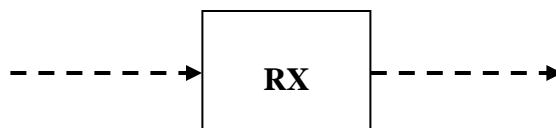


Figura 27. Representación Receptor.

Este concepto (RX) incluye en su universo de atributos el conjunto  $RTR = \{RCP\}$ , siendo que:

Tabla 16. Atributos del concepto receptor

Sigla	Descripción
RCP	Recepción

### 3.3 DIAGRAMA CONCEPTUAL DE LOS NIVELES BAJOS DEL SISTEMA BLUETOOTH.

Con la diferenciación modularizada de los conceptos anteriores, la elaboración del modelo conceptual del sistema transmisor / receptor de Bluetooth consta de tales representaciones las cuales son abstracciones que indican los elementos reales del sistema.

Siendo que se va a representar un enlace completo de Transmisor a Receptor Bluetooth, se tiene que durante la comunicación tales dispositivos pueden ser maestros o esclavos. El maestro es el dispositivo que inicia la conexión a uno o más dispositivos auxiliares. La figura 28 obedece a un diagrama formal de conceptos de primer nivel, el cual es una generalización de todo el sistema y representa el puente de comunicaciones entre el transmisor (maestro) y el receptor (esclavo) para la transmisión de señales N.

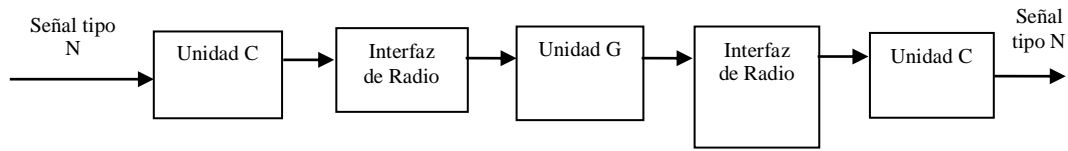


Figura 28. Comunicación entre el transmisor y el receptor.

Esta figura permite tener una visión global del sistema y permite vislumbrar el esquema completo.

**3.3.1 Diagrama completo y el diagrama alternativo de secuencias del sistema Bluetooth.** Después de haber obtenido los conceptos necesarios, los cuales están especificados en la norma IEEE 802.15.1 y llevados a un nivel de abstracción suficientemente global, el modelo genérico propuesto queda conformado por el modulo de transmisión (Figura 29), el de recepción (Figura 30) y el modelo completo (Figura 31), de la siguiente manera:

### 3.3.1.1 Modulo Transmisor ( TX )

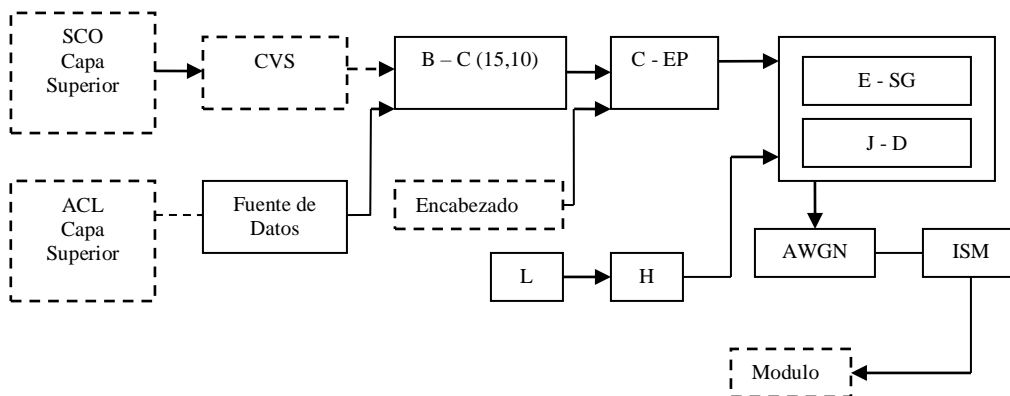


Figura 29. Especificación del transmisor genérico.



### 3.3.1.2 Modulo Receptor ( RX )

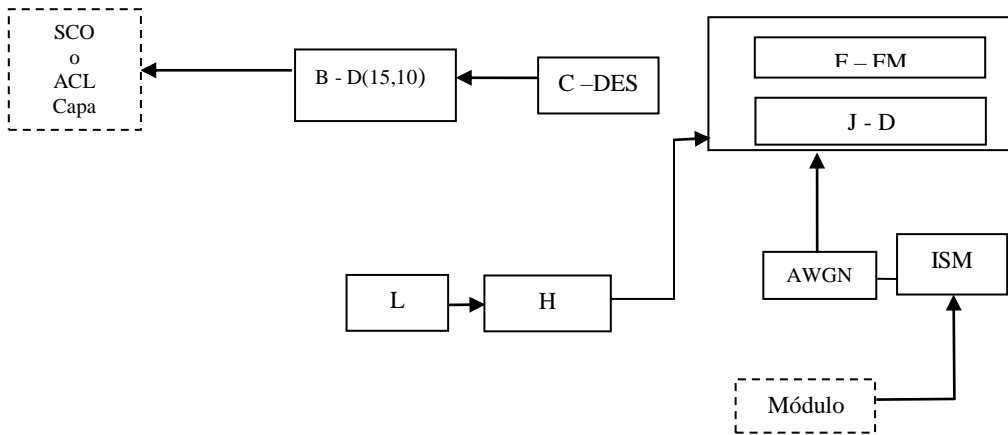


Figura 30. Especificación del receptor genérico.

### 3.3.1.3 Diagrama General (Incluye tipos de señal)

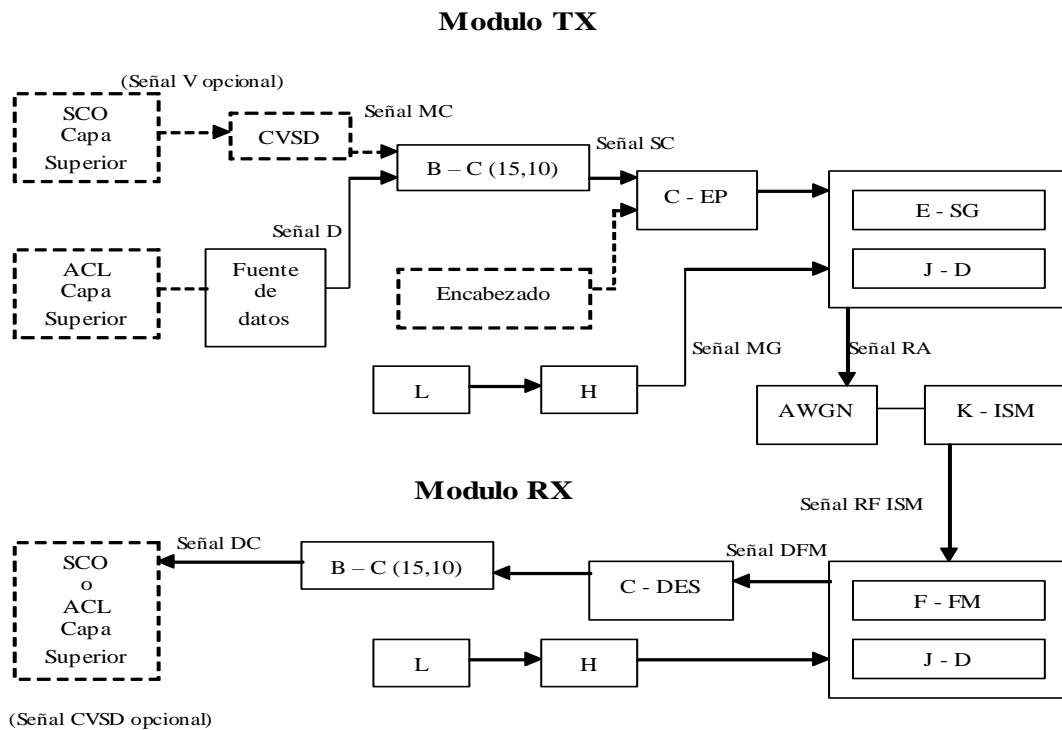


Figura 31. Modelo formal de conceptos completo.

**Nota:** Los módulos en línea discontinua al igual que las señales, representan condiciones opcionales para el diseño final, en esa medida el modelo funcional de simulación trabajara con tipos de señal de datos como entrada fundamental

### **3.3.2 Diagrama de secuencias Bluetooth basado en las asociaciones**

Además de investigar sobre los conceptos del sistema y su estructura, también es preciso investigar en el Diseño de Alto Nivel sobre el comportamiento del sistema, visto éste como una caja negra. Una parte de la descripción del comportamiento del sistema se realiza mediante los Diagramas de Secuencia del Sistema. Este tipo de diagrama emplea procedimientos conocidos también como casos de uso los que representan una interacción genérica. Una instancia de un caso de uso se denomina escenario, y muestra una ejecución real del caso de uso, con las posibles bifurcaciones y alternativas resueltas de forma particular. Para la representación de los procedimientos o casos de uso es más conveniente emplear un estándar de notación más universal. Teniendo en cuenta las asociaciones de conceptos determinadas con anterioridad, se pueden obtener los procedimientos involucrados en el proceso de comunicación del sistema Bluetooth que conforman los casos de uso para el diagrama de secuencias. A manera de ejemplo, una representación de las asociaciones del sistema es el propuesto en la figura 32 (obsérvese que los elementos que contiene hacen referencia a las representaciones propuestas por el grupo de trabajo del Lenguaje de Modelado Unificado (*Unified Model Language*, UML)):

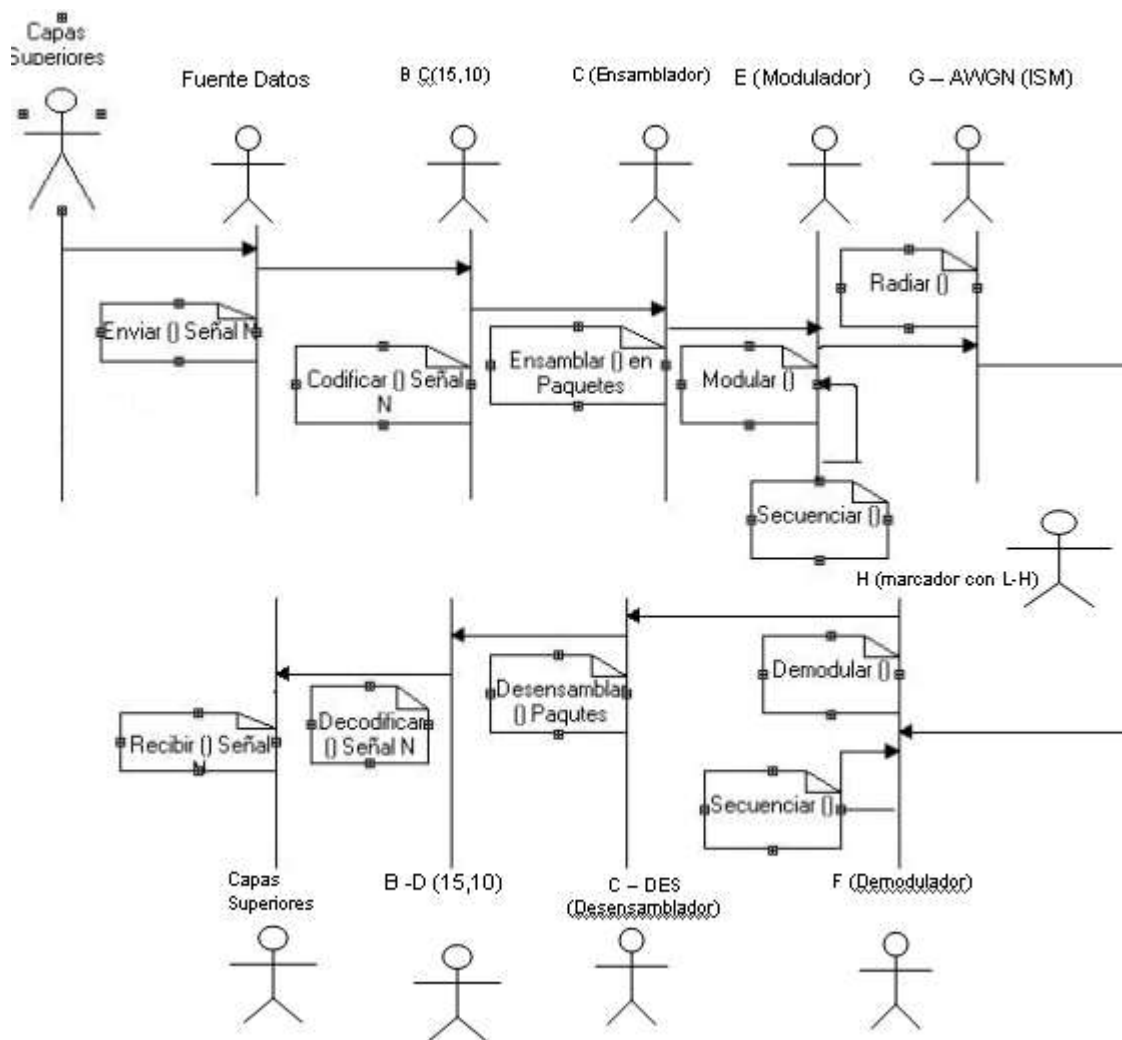


Figura 32. Diagrama de secuencias del sistema TX y RX Bluetooth

### 3.3.3 Procedimientos del Diagrama Secuencial respecto al Modelo

**Conceptual** Finalmente, se tiene de manera cualitativa el significado de los procedimientos:

El cuadro es un elemento del modelado unificado para asociar descripciones al diseño del modelo conceptual.

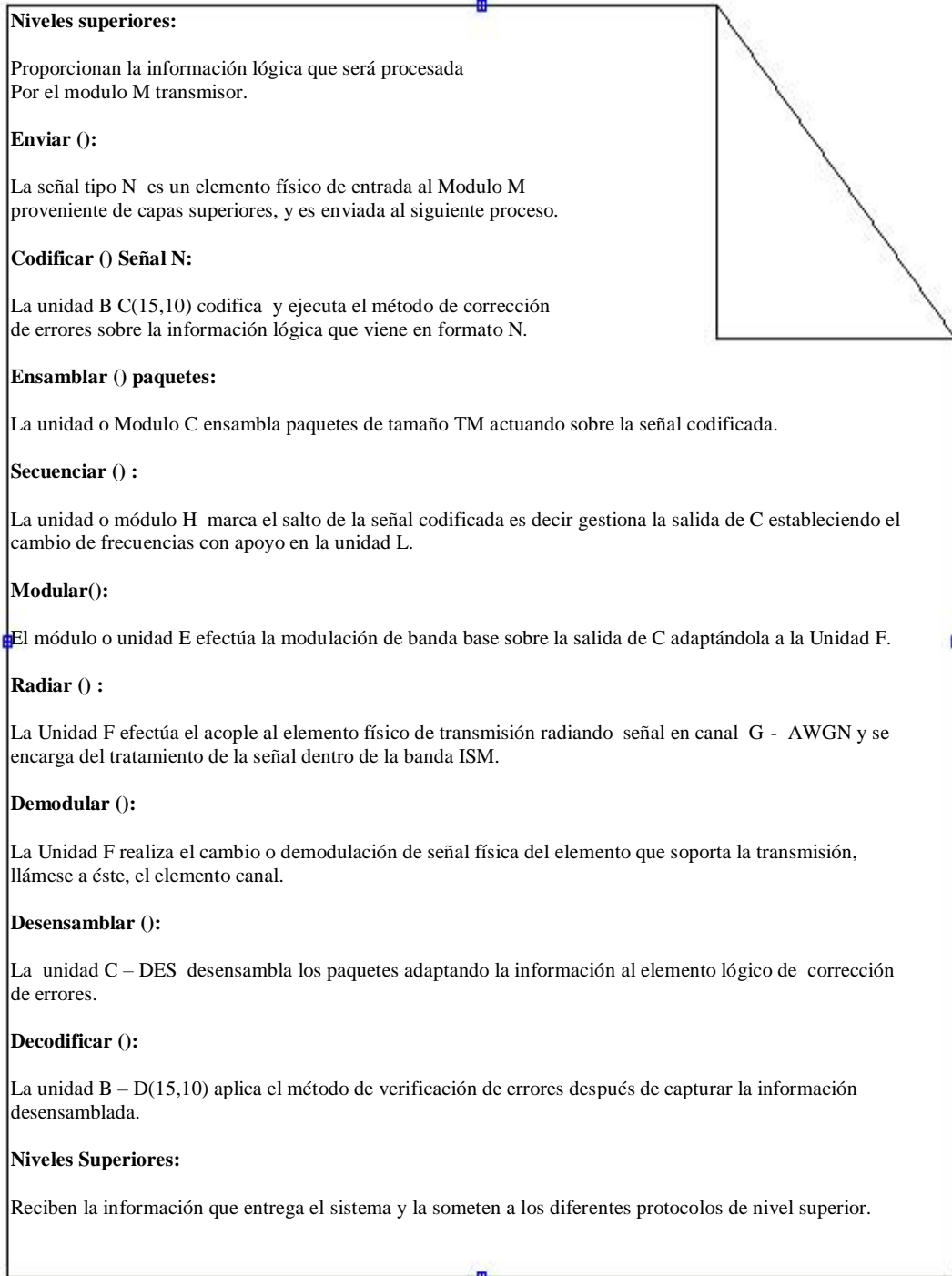


Figura 33. Modelo unificado.

#### 4. ANALISIS DE LA SIMULACIÓN

Lo principal en la configuración es dar valores temporales correctos de acuerdo con la normativa de la IEEE, porque esta da un valor de velocidad en la transferencia de datos para Bluetooth en banda base digital, del mismo modo la norma sugiere cierta velocidad para cambio de canal de frecuencia intermedia que constituye una ventana de tiempo de salto dentro de un rango numérico de posibilidades, para después abarcar la etapa de modulación en banda intermedia y luego la etapa de radio en banda no licenciada. De acuerdo a esto se debe considerar que para lograr la armonización de lo descrito por la norma hay que someter la señal de datos al conjunto de etapas conceptuales tal como la codificación y el proceso de marcado para saltos de frecuencia, el sometimiento al procesador de etapa intermedia de frecuencia y la etapa de radio que debe incluir condiciones del medio o interfaz aérea similares a las reales, descritas dentro de la norma. Para esto es necesario hacer una sincronización de bloques existentes en los toolbox de Simulink® de los que se debe emplear para los modelos conceptuales propuestos en el capítulo III, pero en el marco de la simulación como tal. La cuestión aquí es saber que bloques usar, como configurarlos y lo más crítico: como concatenarlos, dado que existen ciertas deficiencias a la hora de hacer programación rápida asistida por computador, pues no hay muchos módulos

que puedan ser interconectados a otros por incompatibilidad debido al criterio matricial con que han sido elaborados. Por ejemplo unos módulos son basados en bits otros orientados a tramas y aunque el concepto de enlace entre ellos en teoría es sencillo, en la práctica son módulos incompatibles, por tanto el tiempo de diseño se incrementa siendo necesario abortar diseños iniciales o crear módulos complejos propietarios que permitan hacer la comunicación entre dos o más módulos que se esté tratando de interconectar.

Mientras se va trabajando y familiarizando con Simulink<sup>®</sup>, se encuentra que hay mucha desinformación sobre el mismo en medios como Internet lo cual aumenta el nivel de dificultad, porque la ayuda propia del entorno de desarrollo es bastante limitada a la hora de presentar asistencia para los bloques de las cajas de herramientas empleadas para la simulación de Bluetooth, como son la de DSP y la de Comunicaciones. Finalmente se consiguió un manual introductorio de Simulink<sup>®</sup> que se referencia en la bibliografía del trabajo de grado.

A continuación se muestra el seguimiento hecho para el desarrollo del trabajo.

## 4.1 DATOS

Inicialmente era necesario contar con un modo de representar la información dentro de una pico-red. La información consistía en la representación de la misma a manera de datos siendo esta la denominación genérica para la información que proviene de las capas mas altas de la pila Bluetooth que abarca paquetes de datos y voz que de acuerdo con lo propuesto por la norma, estos datos son un flujo de bits con características aleatorias, es decir no programados, para ello se probó el bloque de “fuente aleatoria de datos de Bernoulli”, en ella se configuro una probabilidad de 50% para un bit o muestra simple con valor booleano de 0 y otro 50% para el valor booleano de 1 binario. Esto se puede observar en la caja de configuración de valores de este bloque. Siendo la velocidad digital para Bluetooth de 1 Mbps de acuerdo con la norma, la duda consistía en saber cual sería el tiempo de muestra o bit y la longitud de la ventana de muestras que generase el bloque de fuente de datos. La respuesta está en el tipo de codificación que se emplearía, siendo una de estas la de tipo Hamming (15, 10) donde se ve claramente que la longitud de la palabra ha codificar es de 10 muestras o bits, esta es la longitud de la ventana de datos con la que se configuró finalmente el bloque de Bernoulli. Quedaba faltando el tiempo de muestra o de bit.

Primero se pensó que el tiempo de bit tendría que programarse a 1  $\mu$ s, siendo esto:

$$\begin{array}{l}
 1 \text{ bit} \longrightarrow 1 \mu\text{s} \\
 10 \text{ bits} \longrightarrow x
 \end{array}$$

Donde  $X = 10 \mu\text{s}$ , siendo que:

$$\begin{array}{l}
 1 \mu\text{s} \longrightarrow 1 \text{ bit} \\
 1 \text{ seg} \longrightarrow y
 \end{array}$$

$$y = 1 \cdot 10^6 \text{ bits o } 1 \text{ Mbit},$$

Lo cual conlleva a tener una velocidad de transmisión de  $V = 1 \text{ Mbps}$ . (propuesto por la norma)

Como se puede apreciar se logra obtener la velocidad reglamentaría bajo la norma. No obstante sigue la etapa de codificación que extiende la palabra de datos a 15 bits con los de corrección, por lo tanto ese no puede ser el tiempo de muestra en la ventana de datos.

Se concluyó entonces que como el bloque Hamming extiende la trama a 15 bits y la nueva trama generada duraría el mismo tiempo que la trama de datos de fuente, por lo que se debe programar el bloque de datos con un tiempo correspondiente, así:



1 Mb       $\longrightarrow$       1 seg

15 bit       $\longrightarrow$       Z

*De esa manera Z que es el tiempo que interesa para configurar la fuente de Bernoulli da:*

$$Z = 15 * 10^{-6} \text{ seg} = 1,5 \mu\text{s}$$

Este último valor fue el introducido en el bloque de datos de la fuente de Bernoulli programando de esa manera el tiempo de muestra de manera correcta.

Una vez hecho esto faltaba establecer la compatibilidad con el siguiente bloque a nivel de configuración software de éste.

Simulink sugiere trabajar con datos entramados pues esto aumenta la calidad de la simulación, pero esto va en detrimento del tiempo de simulación que afecta la eficiencia de la plataforma hardware utilizada. Muchos de los bloques empleados solo aceptaban este tipo de configuración y esto dificultó el proceso de implementación del modelo final.

El bloque de datos finalmente se configuró para que su salida fuera en base a tramas con tamaño de ventana de 10 muestras, solo faltaba configurar el siguiente bloque, contando con que este aceptase tanto muestras como tramas.

## 4.2 CODIFICACIÓN

En consecuencia y teniendo presente que la norma considera que dentro de los esquemas de codificación están la FEC 1/3 y el 2/3, se siguió con la recomendación que para datos la FEC 2/3 es el mas adecuado en comparación al clásico Hamming (11,7). Esta FEC se ajusta al patrón de codificación Hamming acortado (15,10) que se emplea, efectuando una expansión de la palabra entrante de longitud de ventana de 10 bits a una de 15 bits, comprimiéndola a su vez para conservar el mismo intervalo de tiempo, lográndose una dispersión o ensanchamiento de bajo nivel.

El Hamming (15,10) trabaja con un polinomio generador que está dado por la norma y es el siguiente:

$$G(X) = (X + 1)(X^4 + X + 1)$$

Al resolverlo el resultado daría:

$$G(X) = X^5 + X^4 + X^2 + X + X + 1$$

Las sumas se calculan mediante suma modulo 2 o del tipo XOR lógica dando como resultado el polinomio compacto así:

$$G(X) = 1X^5 + 1X^4 + 0X^3 + 1X^2 + 1X^1 + 1X^1 + 1X^0$$

$$G(X) = 1X^5 + 1X^4 + 0X^3 + 1X^2 + 0X^1 + 1X^0$$

En esa medida el grupo de coeficientes con los que el codificador debe trabajar queda así:

Coef  $G(X) = 1\ 1\ 0\ 1\ 0\ 1$ , el cual en notación hexadecimal da 65 00x0, este último especificado en la norma.

Este último dato es muy importante pues se considera en la configuración del proceso de codificación.

Al buscar entre los bloques del toolbox de Procesamiento Digital de Señales (*Digital Signal Processing*, DSP), se encontraron varios bloques para efectuar la codificación y entre ellos estaba el codificador y decodificador Hamming (15,10). Al tratar de acoplarlo al bloque de datos de Bernoulli, el problema que se presentó fue la configuración como interprete de tramas y de muestras, por lo tanto se optó por ensayar el funcionamiento del codificador cíclico que estaba dentro de la lista, este permitió la configuración a nivel de tipo transporte (tramas) y se pudo fijar sus parámetro con los datos de longitud de palabra y polinomio generador, así:

K = 10, longitud de la trama de datos de fuente

N = 15, longitud de la trama expandida.

Polinomio Generador = [1 1 0 1 0], como se ve este se introduce a manera de vector fila dado que el bloque lo interpreta de manera correcta como tal por las características matriciales de los objetos de Simulink, de no hacerlo de esa manera el bloque entraba en conflicto con la interpretación del tipo de dato, tomándolo como tipo entero y elemento vectorial con componente única.

La ecuación aplicada por el Hamming es  $2^m - 1$  con  $m \geq 3$  y pertenece a  $+Z$ , esto conlleva a determinar la expansión de la palabra de datos al valor de 16 y por el proceso matemático interno se ajusta a 15, de ahí que sea (15,10).

Para tener una mejor visión del trabajo hecho por la codificación Hamming, obsérvese el siguiente ejemplo con el Hamming de corrección a un bit o Hamming (11,7), En la práctica este modulo FEC tiene utilidad limitada en la transmisión de datos. Se examinará brevemente con el fin de tener un acercamiento con el funcionamiento del codificador Hamming (15,10).

En la teoría de la codificación, el término con que se describe la unidad de mensaje combinada - que comprende los bits de datos útiles y los bits de verificación adicionales - es la **palabra de código**. El número mínimo de posiciones de bit en que difieren dos palabras de código validas es la **distancia Hamming** del código. Por ejemplo considérese el esquema de codificación que tiene siete bits de datos y un solo bit de paridad por cada palabra de código. Si suponemos que se usa paridad par, las siguientes son palabras de código

consecutivas en este esquema:

0000000 0

0000001 1

0000010 1

0000011 0

Se puede ver que un esquema como este tiene una distancia Hamming de 2, pues cada palabra de código válida difiere en por lo menos dos posiciones de bit. Esto significa que no es posible detectar errores de dos bits, pues el patrón de bits resultante (alterado) será una palabra de código diferente pero válida. Se debe aclarar que este esquema simple de codificación incluye solo estas 4 palabras. Sin embargo, sí se detectan todos los errores de un solo bit, ya que si solo se altera un bit de una palabra de código, se producirá una palabra de código no válida.

En general, las propiedades de detección y corrección de errores de un esquema de codificación están relacionadas con su distancia de Hamming. Es posible que para detectar  $n$  errores se requiere un esquema de codificación con una distancia Hamming de  $n + 1$ , y que para corregir  $n$  errores es necesario un código con una distancia Hamming de  $2n + 1$ .

El código con corrección de errores más simple es el código Hamming de un solo bit. Este código no solo detecta la presencia de errores de un solo bit en una

palabra de código recibida, si no también la posición del error. La palabra de código corregida se obtiene complementando el bit erróneo identificado.

Este tipo de código se denomina **código de bloque** porque el mensaje original a transmitir se trata como un solo bloque (trama) durante el proceso de codificación y la subsiguiente decodificación. En general, en los códigos de bloque, cada bloque de  $k$  dígitos fuente se codifica a modo de producir un bloque de  $n$  dígitos ( **$n$  mayor que  $k$** ) de dígitos de salida. Se dice que el codificador produce un código de  $(n,k)$ . La razón  $n/k$  es la **tasa de código** o **eficiencia del código**, y la diferencia  $1 - k/n$  recibe el nombre de **redundancia**.

A fin de ilustrar lo anterior, considérese un código Hamming para detectar y corregir errores de un solo bit suponiendo que cada palabra de código contiene un campo de datos de 7 bits -un carácter ASCII por ejemplo-. Un esquema de codificación así requiere 4 bit de verificación ya que, en este esquema, los bits de verificación ocupan todas las posiciones de bit que son potencias de 2. Se dice que este es un código bloque  $(11, 7)$  con una tasa de  $7/11$  y una redundancia de  $1 - 7/11$ . Por ejemplo las posiciones de bit del valor 1001101 son las siguientes:

---

Posiciones:	11	10	9	8	7	6	5	4	3	2	1	
Bits	:	1	0	0	X	1	1	0	X	1	X	X

---

Las cuatro posiciones de bits marcadas con **X** están marcadas por bits de verificación, los cuales se obtienen como sigue. Los números binarios de cuatro bits correspondientes a las posiciones de bit que tienen un 1 binario se suman empleando aritmética modulo 2 (operaciones XOR), los cuatro bits de verificación son la siguiente suma de 4 bits:

$$11 = 1\ 0\ 1\ 1$$

$$7 = 0\ 1\ 1\ 1$$

$$6 = 0\ 1\ 1\ 0$$

$$3 = 0\ 0\ 1\ 1$$

-----

$$1\ 0\ 0\ 1$$

-----

Así la palabra de código transmitida es:

11 10 9 **8** 7 6 5 **4** 3 **2** **1**

1 0 0 **1** 1 1 0 **0** 1 **0** **1**

En el receptor se efectúa una operación similar: se suman los números binarios de cuatro bits que corresponden a las posiciones de bit que tienen un 1 binario, incluyendo los bits de verificación. Si ni hubo errores, la suma modulo 2 será cero:

$$11 = 1\ 0\ 1\ 1$$

$$8 = 1\ 0\ 0\ 0$$

$$7 = 0\ 1\ 1\ 1$$

$$6 = 0\ 1\ 1\ 0$$

$$3 = 0\ 0\ 1\ 1$$

$$1 = 0\ 0\ 0\ 1$$

-----

$$0\ 0\ 0\ 0$$

-----

Consideremos ahora un error de un solo bit. Supongamos que el bit 11 se altera y cambia de 0 a 1. La nueva suma modulo 2 será:

**Nota:** Obsérvese que en consecuencia el receptor ignorará la operación respectiva sobre la posición 11.

$$8 = 1\ 0\ 0\ 0$$

$$7 = 0\ 1\ 1\ 1$$

$$6 = 0\ 1\ 1\ 0$$

$$3 = 0\ 0\ 1\ 1$$

$$1 = 0\ 0\ 0\ 1$$

-----

$$1\ 0\ 1\ 1$$

-----



En primer lugar la suma es distinta de cero, lo que indica un error; segundo la suma modulo 2, equivalente a 11 decimal, indica que el bit es el bit erróneo. Este bit se invertirá para obtener la palabra de código corregida y de ella los bits de datos correctos.

También puede demostrarse que si ocurren dos errores de bit la suma modulo 2 será distinta de 0, lo que indica un error, pero no será posible determinar la posición del error a partir de la suma. El código Hamming de un solo bit permite corregir errores de 2 bits, pero los demás errores de múltiples bits pasan inadvertidos.

El principal tipo de errores que se presentan en muchas redes de Telecomunicaciones y las redes WPAN como es el caso del redes Bluetooth no son la excepción, consiste en ráfagas de errores, mas que en errores aislados de uno o dos bits. Por ello, aunque el esquema de codificación de Hamming no parece ser el más apropiado para tales redes, a menudo se utiliza esta técnica sencilla para extender el uso de ese esquema.

Para este modulo, como ya se había mencionado con anterioridad, se encontró el correspondiente bloque dentro de la caja de herramientas DSP de Simulink configurando el bloque de Codificación Cíclica como Hamming acortado (15,10), en el cual el tiempo de ventana para una trama expandida de datos quedó fijado a 15  $\mu$ s, es decir:

Este módulo quedó configurado para conservar el tiempo de ventana de la trama proveniente de la fuente de datos, la palabra comprimida en tiempo de muestra quedó fija en 15 bits por trama y el polinomio generador quedó fijado al valor ([110101]) expuesto anteriormente, de acuerdo a la siguiente explicación matemática.

10 bits     $\longrightarrow$     15  $\mu$ s            Tiempo de muestra de la fuente de datos.

1 bit         $\longrightarrow$     X                            Valor de tiempo de muestra.

$$X = 1 \text{ bit} * 15 \mu\text{s} / 15 \text{ bits}$$

*X = 1,5  $\mu$ s es la duración de bit de fuente de datos el cual hay que ajustar para la ventana extendida de Hamming.*

La razón que se tomó para lograr la velocidad de 1 Mbps de acuerdo con el tiempo de muestra de la ventana de fuente de datos, fue:

1,5  $\mu$ s         $\longrightarrow$     1 bit

X               $\longrightarrow$     15 bit

Esto dio como resultado que el tiempo de muestra después del codificador Hamming es de 1  $\mu$ s, de esa manera la velocidad digital quedó en  $V = 1000000 \text{ bps} = 1 \text{ Mbps}$  igualando la de la norma.

### 4.3 ENTRAMADO

Una vez conformados los módulos antes analizados, el problema era abordar la condición de saltos de frecuencia mediante los cuales garantizar el cumplimiento de dicha propiedad de Bluetooth descrita en la norma, que exige una velocidad de 1600 saltos/segundo dejando así un tiempo de 625  $\mu$ s dedicados a un canal. Este intervalo de tiempo es la ventana dentro de la cual tendrían que transferirse las tramas definitivas para pasarlas a la etapa de FI.

De esta manera se concluyó que era necesario armar tramas más grandes para ser transferidas en cada salto de frecuencia, así:

$$1600\_saltos \rightarrow 1\_seg$$

$$1\_salto \rightarrow X$$

$$X = 625\_ \mu s$$

Este último valor debe corresponder con el tiempo necesario para crear tramas de una longitud tal que cumplan con la regla de 1 Mbps como velocidad de transmisión de Bluetooth esto hace que el tiempo de muestra o bit de sea de 1 $\mu$ s. Al tener una trama de tamaño K, esta debería ser transferida dentro de un canal de salto, de esa manera la cuantificación sería:

15\_bit  $\rightarrow$  1\_subtrama, con K = 15 de trama Hamming , pero cuantas de estas subtramas habrá en 625  $\mu$ s?

Luego:

$$15\mu s \rightarrow 1\_subtrama$$

$$625\mu s \rightarrow X$$

$$X = 41,666\_subtramas$$

Pero cada subtrama tiene 15 bits por tanto 41,666 \*15 bits da una longitud de trama de 625 bits, esta debía ser el tamaño final de la trama que quedaría lista para el siguiente modulo. Para corroborar esto, se cuenta una velocidad constante de 1 Mbps = 15 bits/15  $\mu$ s, así:

$f(v,t) = vt$ , con  $v = 15 \text{ bits}/15 \mu s$ , el tiempo para obtener una trama de 625 bits será de 625  $\mu$ s cumpliendo con ello el tiempo de salto. Este tiempo corresponde a un canal de transferencia que tendrá una portadora aleatoria dentro de 79 posibilidades de acuerdo con la norma.

Al buscar en los módulos de Simulink el que en definitiva podía servir para hacer lo anterior era el Buffer perteneciente al toolbox de DSP, el cual quedo configurado así:

Longitud de trama = 625 bits

El buffer lo que hace es tomar un grupo de elementos que para el son muestras pero que del codificador vienen como tramas de longitud de 15 bits y los empaqueta en vectores columna de 41,666 elementos para un total de 625 bits, la salida conserva la velocidad de 1 Mbps pues el paquete armado con 41,666 elementos demora en salir 625  $\mu$ s del buffer concordando con la velocidad de 1600 saltos/seg del marcador de cambio de canal de Bluetooth.

#### **4.4 SALTO DE FRECUENCIA**

En cuanto al marcador de saltos, éste se conformó mediante una función propia de Matlab llamada `RANDINT(X,Y,Z)`, cuyo rango matemático es una matriz de valores que pertenecen a los enteros generados de manera aleatoria, donde:

X = Numero de Filas de la Matriz de salto

Y = Numero de Columnas de la Matriz de salto

Z= Rango de valores dentro de los la función opera de manera aleatoria.

Este módulo se generó mediante la programación de un bloque dedicado para funciones definidas por el usuario mediante el que es posible ejecutar cualquier función de Matlab.

Los valores para la función fueron  $X= 1$ ,  $Y= 79$  que expresa el número de canales en el caso de la norma Norteamericana y  $Z = \{ z/z \text{ pertenece a enteros } > 79\}$ , este conjunto se fijó a 80.

La salida del bloque fue la función aleatorizada:

$$f(X,Y,Z) = \{ f/f \text{ pertenece a enteros dentro de 80 posibles valores} \}$$

Este bloque dio la posibilidad además de programar la ventana de salto siendo esta de 625  $\mu\text{s}$  como resultado de la razón 1/1600 cumpliendo con la norma.

Este bloque debía proporcionar el tiempo de sincronización entre módulo de canales de FI del transmisor maestro y el receptor esclavo, por lo tanto se resolvió conectarlo al módulo de modulación en banda intermedia para cumplir con el requisito de salto de canal, esto garantizaría que no había pérdida de sincronía para evitar errores de bit por este lado.

#### **4.5 MODULACIÓN GFSK Y MEZCLA CON CANALES DE FI**

Siendo que el marcador de saltos quedó parametrizado con 79 posibles valores había a su vez que buscar un módulo de FI que recibiera los valores de salto y los interpretase como valores de portadora de FI de esa manera se pudo lograr

mediante un bloque del set de Comunicaciones de Simulink que permitía la entrada de M valores de portadoras conocido como bloque M-FSK donde la salida del mismo es una señal modulada digitalmente en frecuencia con FSK pero al permitir valores de entrada múltiples como indicadores de cambio de portadora se logró el efecto de desplazamiento en frecuencia a través de salto aleatorio. Este bloque a su vez se configuró para tener anchos de banda de canal de FI de 1 Mhz como la norma lo sugiere.

Teniendo listo el bloque anterior la conexión con el de marcación de salto fue un poco complicada por el mismo motivo de las propiedades de entramado y muestreo que exigen los bloques de Simulink. Siendo que los elementos de salida de la función de aleatorización de salto arrojaba valores enteros, el bloque de FI se pudo configurar también a ese tipo de valor pero la incompatibilidad que apareció fue el de la naturaleza vectorial de lo que salía de la función de aleatorización y lo que el bloque de M – FSK esperaba le llegase. La solución aquí fue hacer un ‘frameo’ o entramado de los valores enteros de la función de saltos para acoplarla con la entrada del M – FSK que exigía entramado y no muestreo de los valores de entrada, quedando resuelto el problema mediante un bloque de entramado dedicado que posee el Simulink dentro de sus librerías.

Por otro lado la información proveniente del buffer debía modularse también a frecuencia intermedia dado que la norma pide que se haga mediante G-FSK para garantizar calidad de la integridad de la información. Al buscar dentro del set de bloques de Comunicaciones no se logró ubicar uno dedicado a hacer este tipo de

modulación digital, sin embargo y siendo que el principio de FSK es el cambio de fase o frecuencia por señal continua para la representaciones de los datos digitales, se optó por probar con el Modulador CPM que permite ser configurado de muchas maneras, donde la modulación por cambio de fase continua es el fundamento de modulaciones como FM y FSK. Al buscar en su caja de configuración efectivamente este incluía el efecto gaussiano sobre la señal moduladora para generar una modulada G – FSK y como la señal de entrada es presenta solo dos valores lógicos, el número M que almacena el conjunto de valores de entrada posibles, se fijó a 2. Dado que la norma especifica que para una modulación de datos dentro de un margen tolerable para evitar efectos indeseables de sobremodulación u submodulación, el índice de modulación debe estar dentro del rango [0.2, 0.35] el valor asignado fue de 0.3, para posteriormente parametrizar el BT o tiempo de ancho de banda a 0.5 que también está especificado en la norma. El bloque fue conectado al ensamblador de tramas (buffer) de tramas de 625 bits no presentando ningún conflicto a nivel de acople.

Solo quedaba tomar los módulos de M-FSK y G – FSK y mezclarlos, donde el primero trabajaba sobre la modulación de 79 posibles portadoras y el segundo el que modulaba los datos bajo FSK gaussiano por norma, esto se hizo utilizando un bloque ‘producto’ que simulaba el mezclador cuya salida arrojó efectivamente, los datos anteriormente entramados y desplazados en frecuencias de las portadoras de salto programadas en la función RANDINT de Matlab. Esta información se paso a un canal de ruido para representar condiciones reales del medio empleando un canal con ruido gaussiano.



#### 4.6 DEMODULACIÓN DE FI, DESENTRAMADO Y DECODIFICACIÓN

Al tomar la señal de entrada al receptor de esclavo, esta debía ser nuevamente normalizada sin canales de salto, para esta parte había que de nuevo tomar el marcador de salto conectarlo de nuevo a un bloque de M-FSK con el fin de lograr la sincronización en la extracción de la portadora aleatorizada que en el tiempo instantáneo  $t$  estaba operando, la salida de este módulo se entregó a un bloque que aplicaba el conjugado a esa portadora dejando su fase en 0, se efectuó la mezcla de la señal de datos con la salida del bloque de calculo de conjugado anclando finalmente los datos a una FI fija. Posteriormente se regeneró la información a su forma digital mediante un demodulador M-FSK, luego se desentramó y decodificó efectuando el proceso inverso al hecho en la etapa de transmisión.

## **5. APLICACIÓN DE LA METODOLOGÍA DE SIMULACIÓN ADAPTADA A MATLAB®.**

En este capítulo se indicarán los módulos utilizados en Simulink® para la realización de la simulación, luego de haber construido el modelo conceptual, trabajando de manera coherente y buscando delimitar la labor planificada dentro del método de diseño software, luego de pasar por las etapas de documentación, planeación y diseño (modelo conceptual), finalizando con la etapa de implementación. Adicionalmente se analizan conceptos y otros aspectos teóricos necesarios para la comprensión del proyecto.

En síntesis, en los siguientes apartados se consolidan definitivamente los métodos empleados tanto a nivel de simulación de sistemas de telecomunicaciones como de diseño de software y se obtiene el modelo de simulación propuesto para el sistema de transmisión y recepción Bluetooth mediante una aplicación implementada en el entorno Simulink® de Matlab®.

## 5.1 CONCEPTOS

**5.1.1 Sistema:** Es un conjunto de partes, entre las cuales se establece alguna forma de relación que las articula. Los sistemas dinámicos son aquellos en los que el tiempo juega un papel fundamental debido a que algunos de sus componentes varían con relación a éste, dentro de estos se encuentran los sistemas de telecomunicaciones y particularmente las redes de área personal inalámbricas (*WPAN*) como las basadas en Bluetooth. [10]

**5.1.2 Simulación** Simular consiste en emplear un modelo con el objetivo de proporcionar una medida aproximada de ciertas variables, esperando que en un ambiente real se obtengan variaciones similares [10]. Con relación a lo anterior para realizar una simulación se establece un modelo virtual y el valor de sus variables de entrada en cada instante de tiempo. Es así que empleando procedimientos o métodos de programación soportados en lenguajes informáticos o empleando otros medios se obtienen entonces las trayectorias de las salidas del sistema, siendo estas últimas las variables obtenidas como consecuencia de parámetros de entrada a través de un grupo de elementos que se relacionan e interactúan entre si y que constituyen un sistema. En tal caso el modelo será útil en cuanto estas trayectorias sirvan para el propósito particular del usuario.

En la actualidad existen muchos ejemplos conocidos de simulación como es el caso de los simuladores de vuelo o la denominada realidad virtual. La ingeniería de simulación permite comprobar algunas características de los equipos diseñados antes de construirlos. En Medicina, Astronomía y en general en la gran mayoría de las ramas de la ciencia, la simulación permite decidir si los modelos manejados se aproximan lo suficiente a la realidad observada. En disciplinas tales como la Ecología o Demografía, los modelos permiten predecir ciertas variables como la población, el nivel de concentración de la capa de ozono, etc. considerando ciertas hipótesis que buscan ser demostradas. En enseñanza la simulación es un medio potente y atractivo para proporcionar al alumno experiencias a un menor costo que los laboratorios tradicionales.

## **5.2 APLICACIÓN DE LA METODOLOGÍA DE SISTEMAS DE TELECOMUNICACIONES ADAPTADA A MATLAB®**

Una vez obtenido el modelo conceptual del sistema Bluetooth, obedeciendo esto al objeto que se va a simular, se aplican los pasos necesarios para enmarcar el modelo conceptual dentro del software de Matlab®, siendo estos:

- Recolección y procesamiento de datos.
- Formulación del modelo conceptual como tal.

- Evaluación del modelo y los parámetros estimados.
- Elaboración de la simulación.
- Diseño de pruebas y validación de la simulación.
- Análisis de los datos obtenidos de la simulación.
- Documentación e implementación de resultados.

**5.2.1 Formulación del problema y plan de estudios** Respecto al estudio de la simulación del sistema Bluetooth se consideró el funcionamiento del sistema el cual se halla en la normativa IEEE 802.15.1 y en el libro del SIG de Bluetooth, cubriendo con esto la necesidad planteada por la metodología de simulación de sistemas de telecomunicaciones acerca de la adquisición de información concreta sobre el objeto de simulación. La documentación relacionada se puede descargar del sitio oficial de la IEEE y el libro de Bluetooth del sitio oficial de Bluetooth.

Dentro de los objetivos de la simulación se encuentran:

- Verificación del cumplimiento de la norma IEEE 802.15.1 con miras al estudio conceptual del sistema Bluetooth desde el punto de vista de los dos niveles inferiores de la pila OSI.

- Comprobar la eficacia de las herramientas ofrecidas por el entorno de desarrollo científico de Matlab® específicamente las toolbox (cajas de herramientas) de Simulink® en cuanto al diseño de sistemas de comunicaciones.
- Llevar a cabo el modelamiento del sistema de comunicación de Bluetooth empleando las herramientas de Simulink® de Matlab® para demostración didáctica y científica del funcionamiento del mismo, considerando la ventaja de Simulink® de Matlab® para estudio de sistemas complejos.

En consecuencia a los objetivos anteriores se abordó la solución al estudio de funcionamiento del sistema Bluetooth mediante un modelo de simulación, para lo cual se tienen como objetivos técnicos:

- ◇ Expresar como variables de entrada al sistema, señales de información digital de datos y alternativamente señales analógicas (voz sobre Bluetooth).
- ◇ Mostrar la manera de realizar la digitalización de señales de voz (CVSD) de entrada para adecuarla a una unidad de transmisión Bluetooth (opcional).
- ◇ Efectuar codificación de voz (opcional) o datos de entrada para garantizar integridad de información.

- ◇ Mostrar el mecanismo mediante el cual se efectúa el proceso de Espectro Ensanchado por Saltos de Frecuencia (FHSS), para seguridad de la transmisión y la recepción de la información.
- ◇ Expresar como variables de salida (Modulo Transmisor) la señal modulada para ser radiada.
- ◇ Verificar la integridad de la información mediante un proceso de decodificación y comparación de datos recibidos.
- ◇ Recuperar la señal de voz recibida mediante una decodificación de tipo digital a analógica. (Opcional)

**5.2.2 Recolección y procesamiento de datos** Los datos adquiridos para la elaboración del modelo de simulación se basan en la norma de la IEEE, información teórica obtenida de Internet y modelos parciales encontrados en la misma fuente, estos datos se exponen en el capítulo uno de este documento.

**5.2.3 Formulación del modelo de simulación del sistema de enlace transmisor – receptor, Bluetooth** La formulación del modelo conceptual Bluetooth implicó la aplicación de niveles de abstracción que brindaron la posibilidad de tener una visión real y simplificada de los diferentes procesos que

conforman el sistema en cuestión. Esta tarea se hizo en el capítulo anterior en el que se modeló conceptualmente el sistema, considerando los atributos fundamentales de los procesos inmersos y las interacciones entre estos procesos, expresadas como asociaciones entre conceptos [9].

**5.2.4 Evaluación del modelo y parámetros estimados** La metodología de simulación de sistemas de telecomunicaciones sugiere ajustar los valores a los rangos considerados para las variables de los diferentes procesos, para llegar a modelar adecuadamente el sistema se debe describir correctamente las variables involucradas. Los rangos de las variables y las relaciones entre los diferentes procesos se toman del modelo conceptual, teniendo en cuenta que:

- Los rangos de las variables se ajustan de acuerdo con los bloques propuestos por Simulink® y con ellos se hacen los ajustes de ensayo y error.
- Las relaciones de los procesos o módulos del sistema Bluetooth se toman del modelo conceptual y están plasmados de manera implícita en el modelo de simulación, construido más adelante cuando se muestre en el modelo de bloques de Simulink®.

Además dado que fundamentalmente se trabaja con Matlab®, particularmente con Simulink®, las características de los bloques que presenta este lenguaje de simulación, permiten obviar muchos de los análisis previos de los lenguajes de



propósito general como C, C++, Java, Perl, Python, etc., gracias a esto, el intrincado análisis del desarrollo orientado a objetos o por otro lado el basado en estructuras se ve desplazado y abre paso a la programación mediante bloques visuales propios de las cajas de herramientas de lenguajes de simulación especializados, como es el caso de Simulink®. Esto disminuye en gran medida el tiempo de elaboración del modelo de simulación del sistema de transmisión – recepción de Bluetooth y a su vez disminuye el esfuerzo invertido en el establecimiento de asociaciones y relaciones exigidas entre los objetos de los lenguajes orientados a estos.

Como se había sugerido al principio de este numeral y haciendo énfasis de manera redundante en el ajuste de los rangos de variables, esta tarea se lleva a cabo más adelante en los numerales correspondientes a la elaboración del programa como tal y a sus pruebas de validación, buscando depurar de esa forma el funcionamiento del modelo de simulación final.

No obstante, se proponen valores iniciales de prueba para las variables, en cumplimiento con el numeral actual de la Metodología de Simulación de Sistemas de Telecomunicaciones y que corresponden a los diferentes módulos conceptuales dados en el capítulo anterior, e implementados mediante bloques utilizando la herramienta de simulación Simulink®. El sistema está representado por los siguientes bloques:

- Datos
- Codificación Hamming
- Entramado
- Modulación GFSK
- Saltos de Frecuencia
- Canal de transmisión
- Demodulación
- Desentramado
- Decodificación
- Comparación

Como se mencionó anteriormente se dan valores estimados para los parámetros de los diferentes bloques.

- **Datos:** Valores propuestos de la variable a considerar:

**Señal D: Tipo Datos,** Se toma una secuencia de bits aleatoria con rangos de amplitud 1 y 0.

**Señal V: Tipo Voz o Analógico (Opcional),** Se toma señal analógica tal como una función  $\text{sen}\theta$ , con amplitud normalizada sin tener en cuenta la amplitud real de la señal o señal de fuente estándar. Tal señal codifica un modulador CVSD para digitalizar la señal analógica.

- **Codificación** Valor propuesto de la variable a considerar:

**Señal SC, tipo codificada (15,10)**, Se busca que la señal de salida sea una trama codificada de longitud inicial propuesta de 10 bits y extendida a 15 bits donde se incluyen los bits de codificación para detección y corrección de errores. Así los parámetros propuestos son:

Longitud : 15 bits en codificación , 10 bits de dato original.

Amplitud : de cada bit, 1 o 0 discretos y sin considerar valor real.

- **Entramado:** Valor propuesto respecto a la variable a considerar:

Señal entramada: esto significa que los bits de información codificada salen del modulo en un formato de tramas, enmarcadas en una longitud dada por la norma y a una velocidad tal que cubre el valor propuesto por la misma norma. De esta manera el módulo será parametrizado así:

Señal : Entramada

Tamaño: 625 bits

Velocidad de transmisión de datos: 1 Mbps

- **Modulación GMSK** : Este bloque es el encargado de realizar la modulación a la señal de datos provenientes del modulo anterior y está parametrizada de la siguiente manera:

Señal de entrada: señal M-aria de orden 2, es decir es de tipo binaria con velocidad de 1 Mbps. Efectúa una conversión de señal digital a analógica diferenciando el tipo de símbolo (1 o 0) por medio de dos posibles valores de frecuencia.

Señal de entrada: Digital

Señal de Salida: Analógica

$F_c$ : Frecuencia portadora  $\cong 2,4$  GHz.

$\Delta f$ : Está dentro del rango [115 KHz, 175 KHz].

- **Salto de frecuencia:** Este bloque es el encargado de marcar el paso de salto para las diferentes portadoras, para efectuar el proceso de salto en frecuencia.

Los parámetros de este componente del sistema son:

Señal: generada mediante función aleatoria

Rango de canales: [0, 78]

Rango Portadoras: (2,402 GHz, 2,480 GHz)

Ancho de Banda por Canal de Salto: 1 MHz.

Rango Escala de salto: [- 39, 39] MHz.

- **Canal:** Este bloque representa el medio de transmisión por interfaz de aire y proporciona las propiedades de ruido necesarias lo que se contempla mediante la relación señal a ruido ( $E_s/N_o$ ) y el periodo de símbolo, adicionando al sistema un generador de ruido Gaussiano.

Parámetros estimados:

Es/No: 20 dB

Periodo de Símbolo: 1 MHz.

- **Demodulador:** Este bloque del receptor es el encargado de retornar la señal analógica procedente del canal, a una señal digital de orden M-ario, dado que se necesita una salida binaria entonces el valor de M se asume como 2.

Señal estimada de salida: binaria ( 1 y 0) sin estimación de valores reales para la amplitud.

- **Desentramado:** Este bloque es el encargado de convertir la señal empaquetada en tramas a muestras.

Señal binaria de tramas de 625 bits

Muestras resultantes de 15 bits.

- **Decodificación:** Esta bloque es el encargado de la detección de errores en la señal de información, hacer la corrección de los mismos garantizando la integridad de los datos.

Señal de salida: flujo de bits con una estimación en la corrección por cada 10 bits.

- **Comparación:** Este bloque se encarga de medir el número de errores presentes durante la transmisión de la información, para propósitos de verificación.

Señales de entrada: datos binarios de la fuente y señal procedente del módulo de decodificación.

Señal de salida: números enteros capturados en despliegue que muestra el error resultante.

### **5.3 PROGRAMACIÓN DEL SISTEMA TRANSMISOR – RECEPTOR BLUETOOTH BAJO SIMULINK®**

El sistema de comunicación se puede construir de los bloques encontrados en Simulink® [10] dentro de las categorías de Procesamiento Digital de Señales (*Digital Signal Processing, DSP*) y Comunicaciones (*Communications*). La manera como se construyó el enlace de comunicaciones del sistema completo de transmisión unidireccional Bluetooth en Simulink®, fue trabajando hacia fuera en algunas partes del enlace, por ejemplo se realizó la codificación basada en Hamming (15,10) antes de implementar la fuente de datos binaria encargada de representar el flujo de datos proveniente de niveles superiores, por la razones de requerimientos de Bluetooth explicados en el capítulo anterior, en otras por el contrario si se realizó de acuerdo a la dirección del flujo de datos del enlace, como

es el caso de la modulación el canal, el entramado, etc. De la misma forma se procedió con el lado del receptor.

Tanto el nivel físico como el de enlace de datos, los cuales están ubicados en los dos niveles inferiores de la pila de protocolos del sistema de comunicaciones Bluetooth, pueden ser representados idealmente como diagramas de bloques y esto es porque herramientas como Simulink® de Matlab® se pueden emplear de manera apropiada para desarrollar el modelo. A continuación se detallan los bloques de Simulink® de los respectivos blocksets empleados para la construcción del modelo definitivo.

**5.3.1 Datos** Este bloque representa una fuente de generación aleatoria de bits.

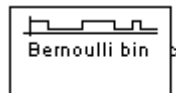


Figura 34. Bloque generador de tren binario aleatorio

Este bloque esta conformado por el Generador Binario de Bernoulli que genera de manera aleatoria números binarios utilizando una distribución Bernoulli. La distribución Bernoulli con parámetro  $p$  produce cero con una probabilidad  $p$  y uno con una probabilidad  $1-p$ . En seguida se muestra la configuración de este bloque.

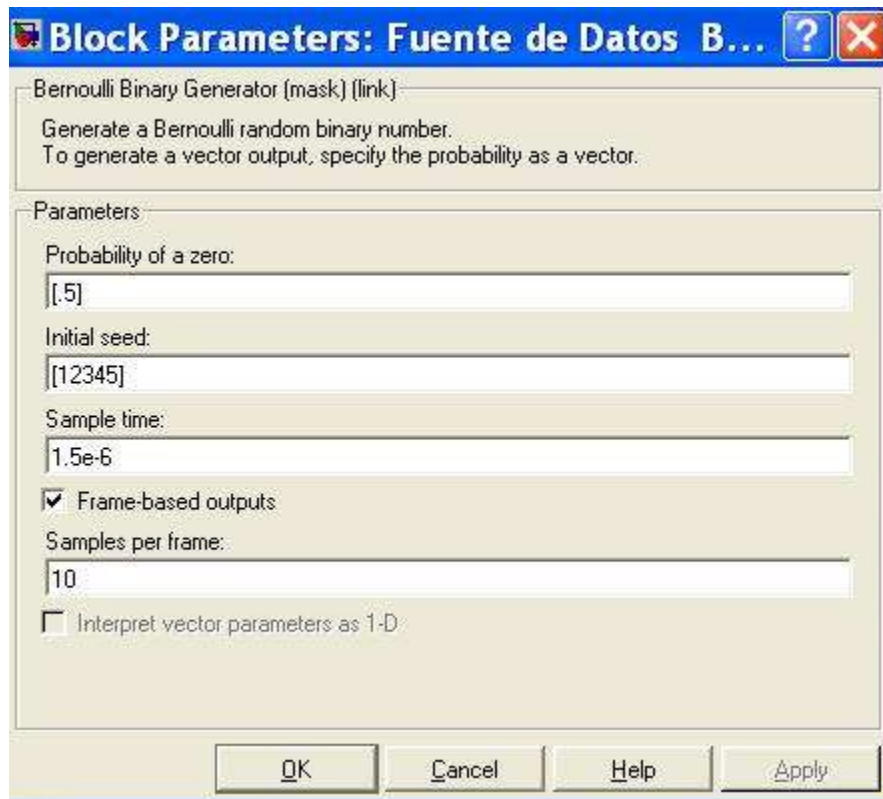


Figura 35. Generador Bernoulli para datos.

Este bloque queda configurado bajo las siguientes condiciones:

El campo de **probabilidad de un cero** se refiere a la probabilidad con la que puede ocurrir un valor lógico de cero (0), siendo esta de 50% de este modo la probabilidad de ocurrencia de un uno (1) esta dado por  $1 - p$  y seria igual a 50%.

El campo de **condición inicial**, se configura con un valor escalar de uno para dar la funcionalidad al generador aleatorio aunque puede ser cualquier vector o escalar.

La casilla de **tiempo de muestra** es el periodo de cada vector basado en muestras o cada fila de una matriz basada en tramas, para el caso el valor de



tiempo por muestra esta fijado a  $1.5e-6$  segundos para garantizar el cumplimiento de la tasa de 1 Mbps para Bluetooth.

La casilla de **salida basada en tramas** es habilitada para configurar el campo de **muestras por trama** que tiene un valor de 10 muestras o bits.

**5.3.2 Codificación** Para la codificación se utiliza el bloque decodificador cíclico binario de Simulink®.

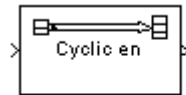


Figura 36. Codificador Hamming.

Este bloque es un codificador cíclico sistemático que recibe un mensaje de longitud  $K$  y genera una palabra codificada de longitud  $N$ . El número  $N$  puede tomar la forma  $2^M - 1$ , donde  $M$  es un entero mayor o igual a 3. La entrada puede contener exactamente  $K$  elementos. Si está basado en tramas, entonces debe ser un vector columna. La salida es un vector de longitud  $N$ . Se puede determinar el esquema de codificación cíclica binaria de dos maneras:

- Crear un código  $[N,K]$ , para ello se configura en la caja de dialogo los valores de  $N$  y  $K$  como primero y segundo parámetros respectivamente. El bloque calcula un polinomio generador adecuado, esto es, `cyclpoly(N,K,'min')`.

- Crear un código con palabra de longitud N y un polinomio generador binario de grado particular (N,K), donde N es el primer parámetro y el segundo parámetro es un vector binario. El vector representa el polinomio generador y los componentes listados son sus coeficientes en el orden de exponentes ascendentes. Esta configuración es la empleada para este módulo del sistema de transmisión Bluetooth y se muestra a continuación:

Donde la casilla de **longitud de palabra codificada N** presenta el valor 15 para la codificación Hamming (15,10) y la casilla de **polinomio generador o longitud de mensaje K** presenta un vector con los valores dimensionales necesarios y característicos para la misma codificación, siguiendo las indicaciones que están dentro de la norma.

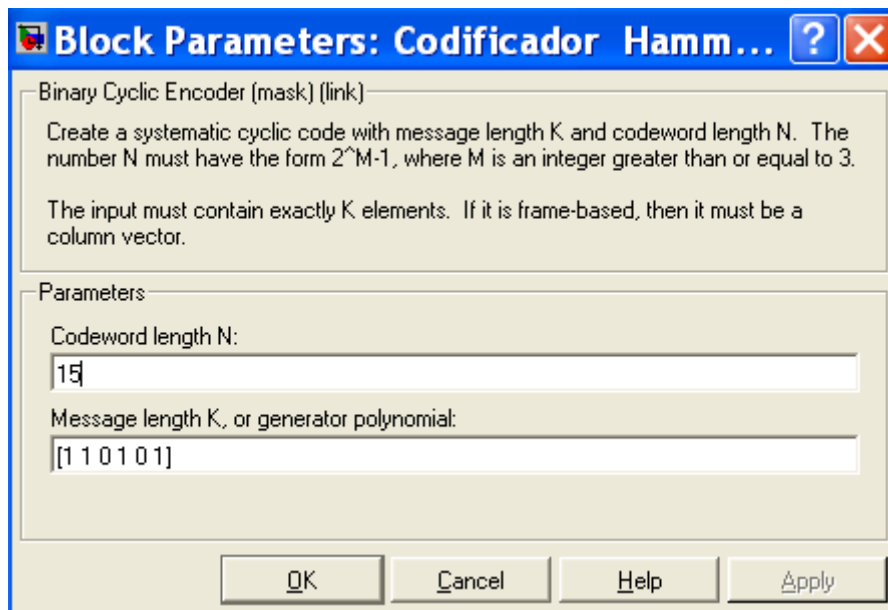


Figura 37. Configuración del Codificador Hamming.

**5.3.3 Entramado** Este bloque esta compuesto por el bloque Buffer de Simulink®

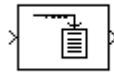


Figura 38. Buffer de Simulink®.

El bloque Buffer redistribuye las muestras de entrada en un nuevo tamaño de trama, mas grande o mas pequeño que el tamaño de la trama de entrada. El almacenamiento de tramas de tamaño grande produce una salida con una tasa de tramas más baja que la velocidad de tramas de entrada, sin embargo para la simulación del sistema Bluetooth es necesaria esta condición para cumplir el requisito estipulado en la norma. La configuración se observa en la siguiente figura:

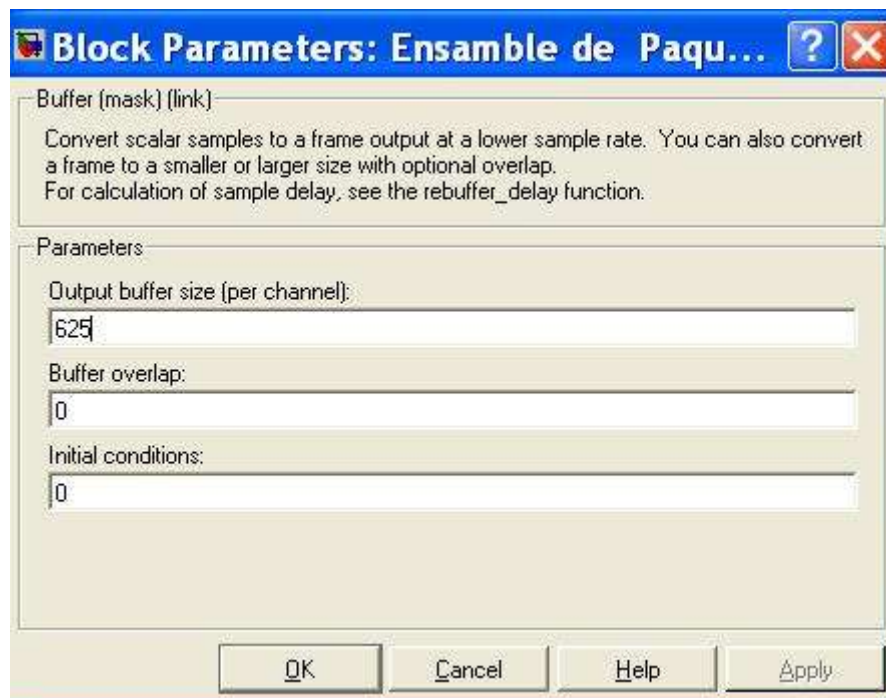


Figura 39. Configuración del buffer para entramado.

**5.3.4 Bloque de Modulación de Frecuencia GFSK** Este módulo es responsable de realizar la interpretación analógica de los bits que llegan a él, llevando la señal analógica de frecuencia intermedia, adicionalmente por medio de un multiplicador efectúa esta operación entre la señal analógica y la señal entregada por el generador de saltos de frecuencia (la cual se lleva al mismo formato de la señal analógica antes mencionada), llevando la señal a los 79 posibles canales de propagación.

Finalmente la señal que está en frecuencias intermedias del orden de los Mhz, es modulada mediante una portadora de alta frecuencia a 2,4 Ghz (banda ISM) permitiendo así cumplir con la recomendación de la norma.

**NOTA:** Para efectos de simulación solo se trabajará a nivel de frecuencia intermedia por limitaciones de hardware del PC dado que al implementar el modulador y demodulador de RF para la banda de 2,4 Ghz, el computador entra en retardos inapropiados para la simulación.

El bloque de Modulador M-FSK modula utilizando el método de cambio de frecuencia de orden M-ario. La salida es una representación de frecuencia intermedia de la señal modulada. El parámetro numérico, M, es el numero de frecuencias de la señal modulada.

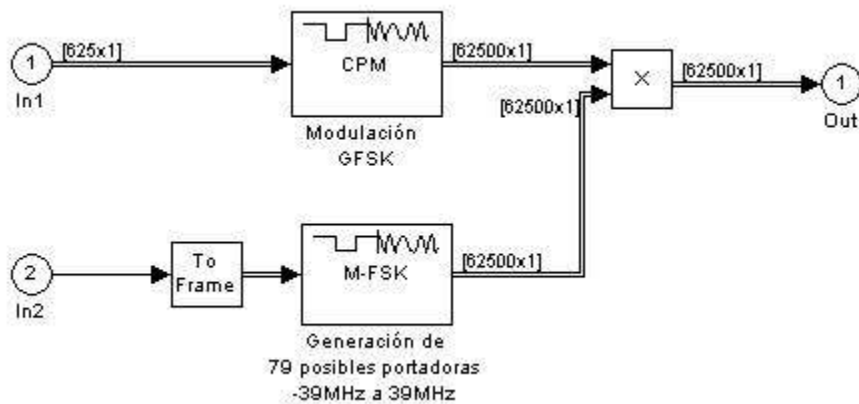


Figura 40. Modelo de Modificador GFSK.

El parámetro de **separación de frecuencia** es la distancia, en Hz, entre frecuencias sucesivas de la señal modulada. Si el parámetro de **continuidad de fase** se fija a continuo, entonces la señal modulada mantiene su fase incluso cuando cambia su frecuencia. Si el parámetro de **continuidad de fase** se fija a discontinuo, entonces la señal modulada comprende porciones de M sinusoides de frecuencias diferentes; así, un cambio en el valor de la entrada podría causar un cambio en la fase de la señal modulada. Este bloque se configura con los siguientes valores:

Como se observa en la siguiente figura, se define el **número M-ario** para un valor de 79, este indica el número de canales de saltos de frecuencia, de acuerdo con la norma, esta configuración se realiza con el fin de llevar los valores enteros entregado por el generador de saltos al mismo formato utilizado por el modificador GMSK.

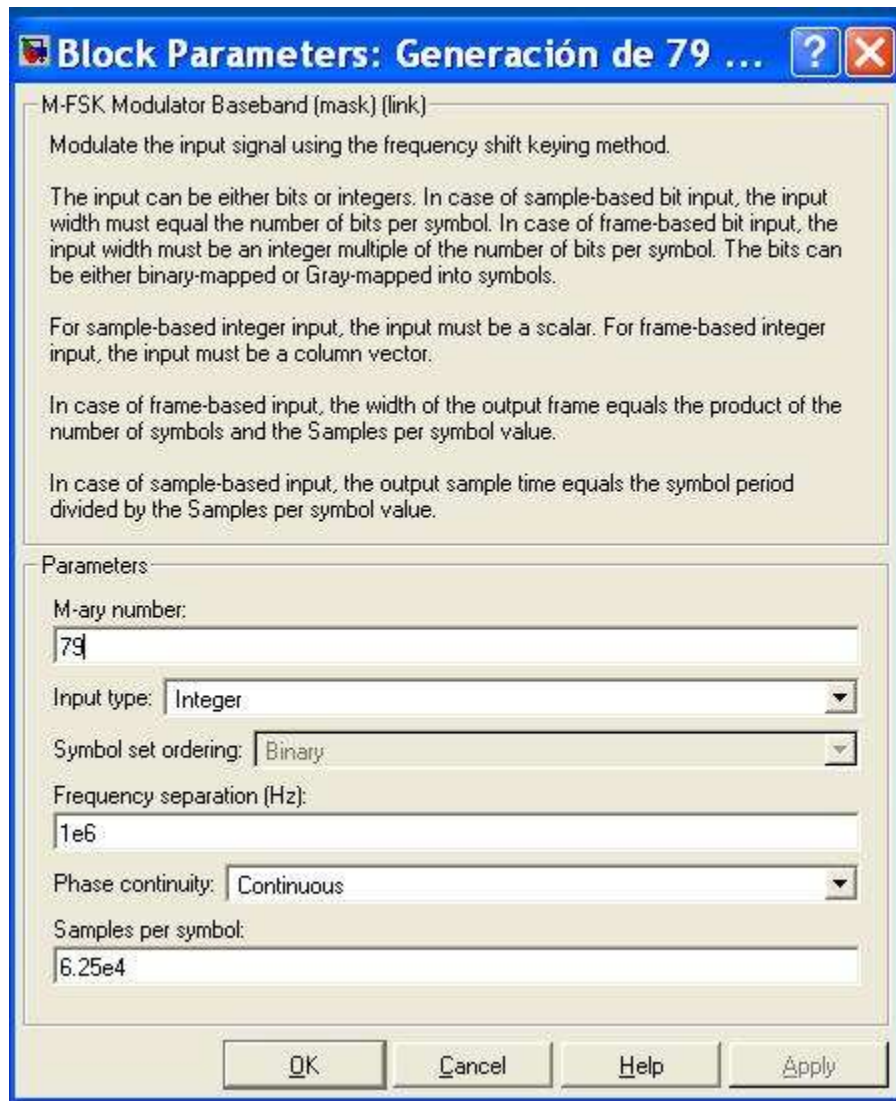


Figura 41. Configuración del Bloque M-FSK.

El bloque producto que se muestra en la figura 40 realiza multiplicaciones o divisiones a sus entradas, en este caso realiza una multiplicación.

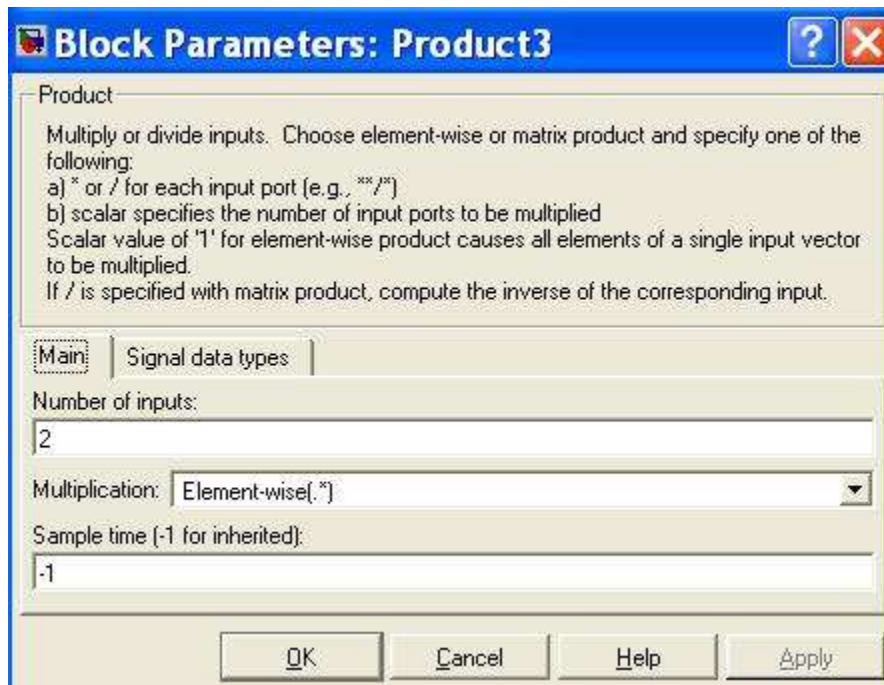


Figura 42. Configuración del bloque producto.

La casilla **Multiplicación** tiene dos opciones, una para multiplicar matrices y la otra para multiplicar vectores o escalares, como la entrada del bloque son dos vectores entonces se configura como Element-wise que es el indicado para este caso, las otras dos casillas son de una configuración orientada por los parámetros de Simulink®.

### 5.3.5 Canal Descripción del Bloque: canal de Ruido Gaussiano

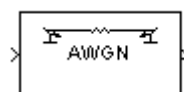


Figura 43. Bloque de Simulink®.

Este bloque agrega Ruido Blanco Gaussiano a la señal de entrada compleja o real. Sí la señal de entrada es compleja, este bloque adiciona ruido Gaussiano complejo y produce una señal de salida compleja. Además emplea el bloque de fuente aleatoria del Blockset de procesamiento de señales para generar el ruido. El parámetro inicial en este bloque inicializa el generador del ruido. El parámetro inicial puede ser un escalar o un vector cuya longitud coincide con el número de canales en la señal de entrada, en este caso se fija a uno (1) como valor escalar por ser canal único.

Como este bloque sirve para simular el canal de transmisión, debe ser construido o programado para diferentes entradas, tal como se presenta en la realidad, por lo que se describirá cuatro de las posibles entradas al bloque y el tratamiento que este le da a cada una de ellas.

- Señal de entrada escalar basado en muestras, entonces el bloque adiciona ruido gaussiano escalar a su señal.
- Señal de entrada vectorial basada en muestras o un vector fila basado en tramas, entonces el bloque adiciona ruido gaussiano independientemente a cada canal.
- Señal de entrada vectorial columna basada en tramas, entonces el bloque agrega una trama de ruido Gaussiano a su señal de canal simple.



- Señal de entrada matriz basada en tramas de orden  $m$  por  $n$ , entonces el bloque agrega una trama de longitud  $m$  de ruido gaussiano de manera independiente para cada uno de los  $n$  canales.

La entrada no puede ser una matriz de orden  $m$  por  $n$  basada en muestras si tanto  $n$  como  $m$  son mayores que uno.

En la siguiente figura, se muestran los parámetros a configurar para la simulación con relación a este bloque:

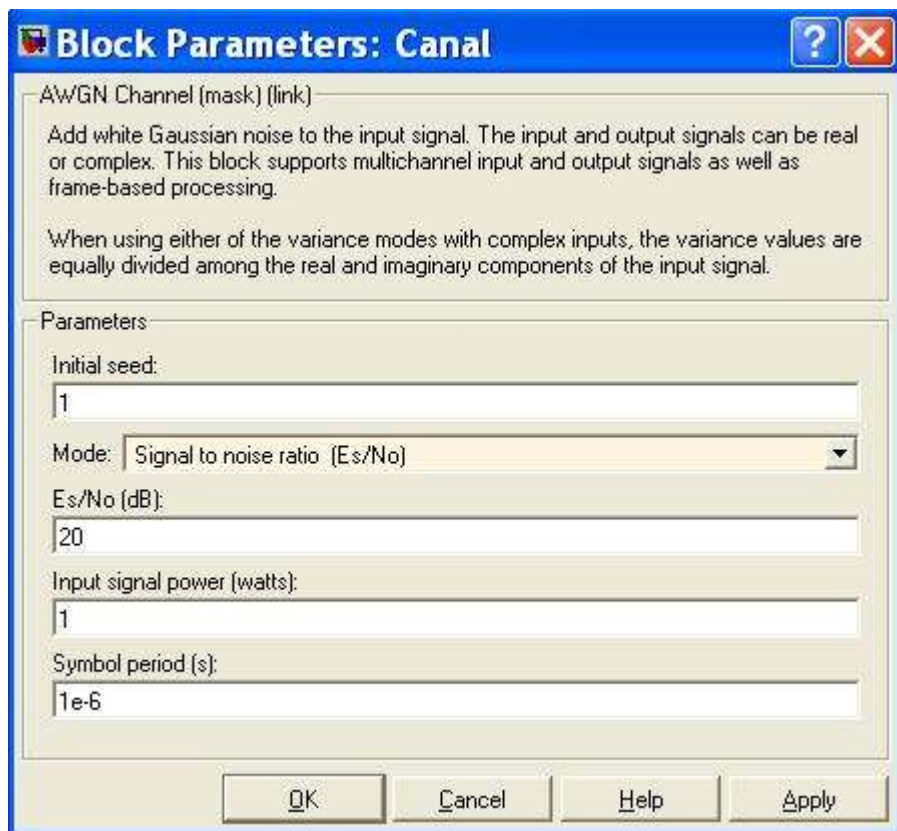


Figura 44. Configuración del canal de transmisión.

**5.3.6 Demodulador** Este bloque está conformado de manera similar al bloque modulador, explicado en el numeral 4.3.2. En la siguiente gráfica se muestra el bloque completo de demodulación que hace parte del receptor, este es el encargado de retornar la señal analógica, que viene por diferentes canales de la banda ISM, a la forma digital en tramas de 625 bits.

Como se mencionó anteriormente la configuración de los bloques pertenecientes a este módulo es idéntica a la vista en la modulación, entonces se explicará únicamente los elementos diferentes, como es el caso del módulo generador del conjugado utilizado para eliminar los saltos de frecuencia.

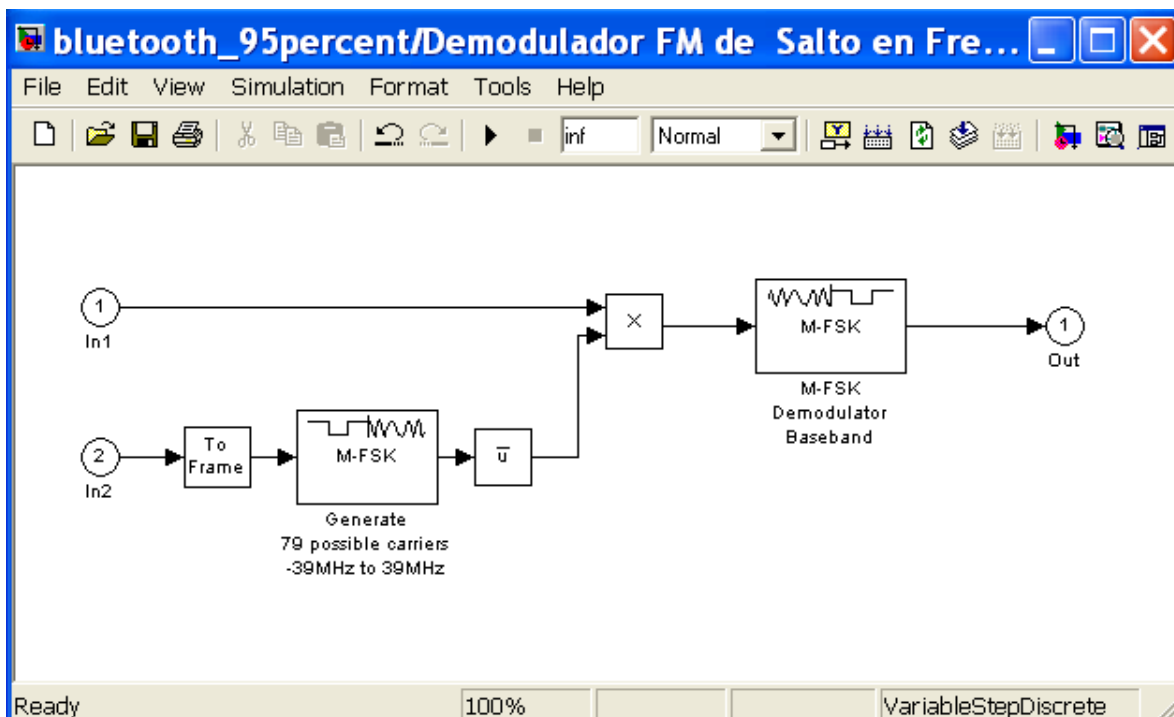


Figura 45. Bloque de demodulación GMSK.

El bloque empleado para generar el conjugado del generador de saltos es el bloque **Función Matemática**, el cual realiza una función matemática.

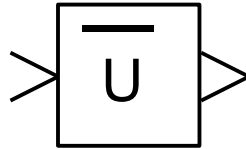


Figura 46. Bloque función matemática.

El bloque de función matemática realiza numerosas funciones matemáticas comunes, las cuales se pueden seleccionar de una lista de posibilidades. Para el caso de la simulación, se selecciona la función conjugada, porque el propósito de este bloque es retornar la señal analógica que viene por 79 posibles canales a las frecuencias original de  $2,4 \text{ GHz} \pm \Delta f$  que es la generada por la etapa de modulación antes de los saltos de frecuencia.

Las entradas para este bloque pueden ser reales o complejas, en la casilla de **Función** se selecciona la función que se requiere, para el particular se selecciona la función conj, la siguiente casilla **Tipo de señal de salida** se selecciona en auto, porque es la que permite salidas de tipo real y complejo simultáneamente. Las otras dos opciones (*Real* y *Complex*) entregan a la salida únicamente valores reales y complejos respectivamente, sin simultaneidad.

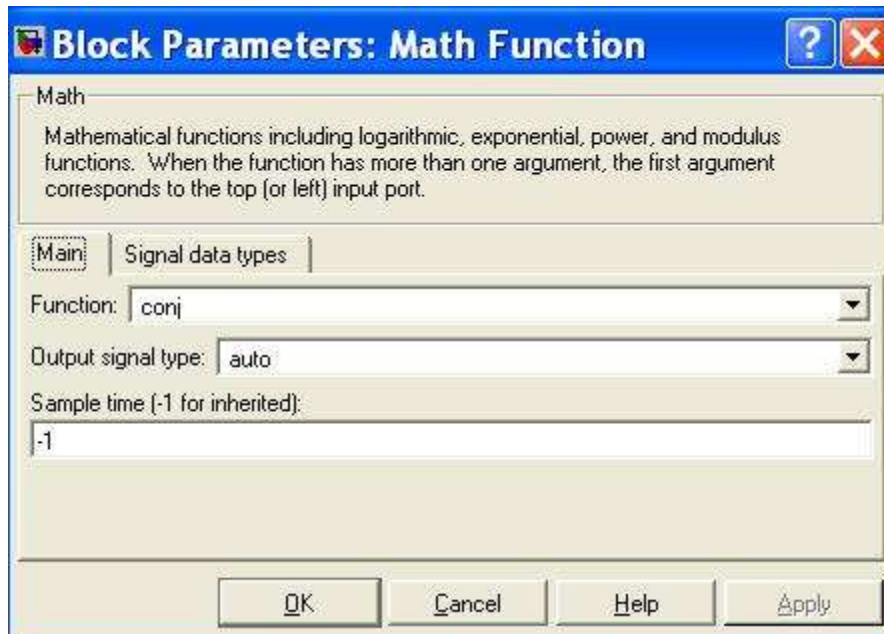


Figura 47. Configuración de la función matemática.

**5.3.7 Desentramado** Este bloque está conformado como en el entramado, por un buffer y se le adiciona a este bloque un módulo de retardo, para efectos de simulación.

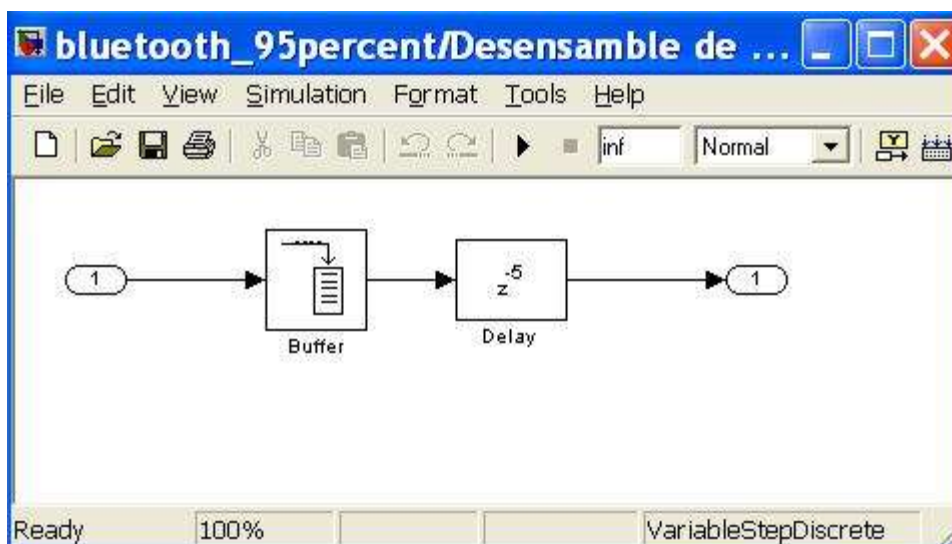


Figura 48. Bloque de desentramado.

El buffer se configura de manera similar al buffer de entramado, la única diferencia es que en este la salida es de 15 bits, como este valor es menor al número de bits que recibe, la condición inicial se deja en valor de cero, por cuanto no tiene que aumentar bits a las tramas. En la figura 49 se ve la configuración del modulo.

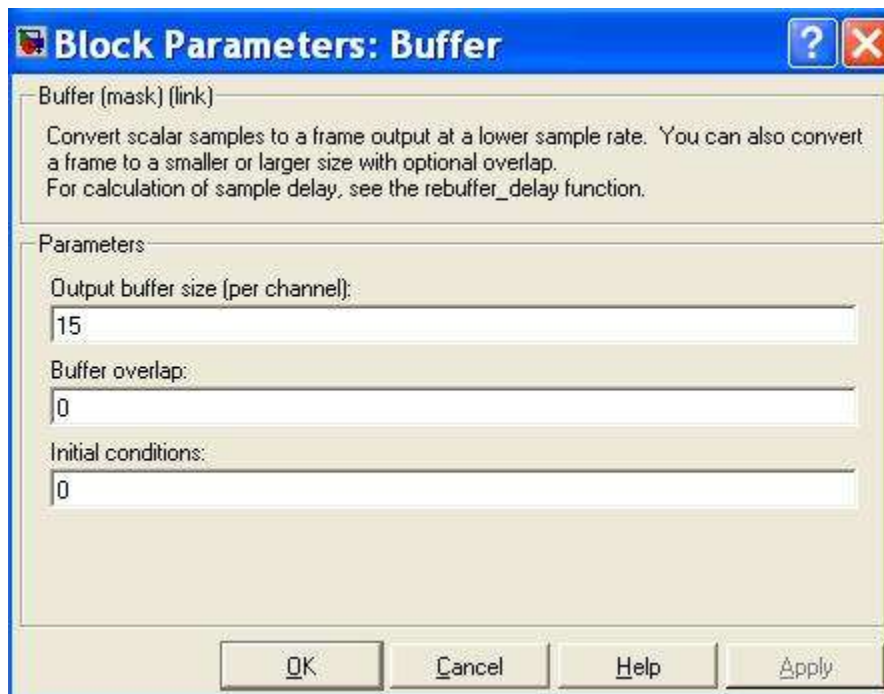


Figura 49. Configuración del buffer del receptor.

El retado introducido en este bloque se hace a través del modulo de Delay de Simulink® y se configura a gusto del programador, como se muestra en la figura 50.

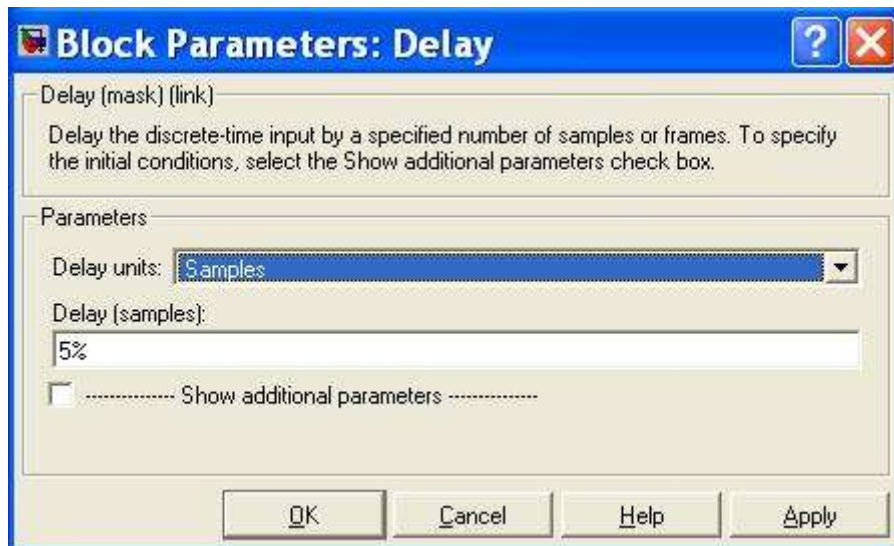


Figura 50. Configuración del retardo.

**5.3.8 Decodificador Hamming** Para la implementación se emplea el bloque decodificador cíclico binario de Simulink®, al igual que en el bloque codificador

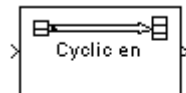


Figura 51. Decodificador Hamming.

La configuración de este bloque es similar al bloque codificador del numeral 4.3.4. por lo cual no se explicaran las casillas de este, únicamente las diferencias.

En este caso las tramas que llegan al bloque tienen una longitud exacta de 15 bits y al retirarse el polinomio generador (adicionado en la codificación) entrega una

muestra de 10 bits al receptor. Con este bloque se puede estimar cuantos paquetes llegaron correctamente.

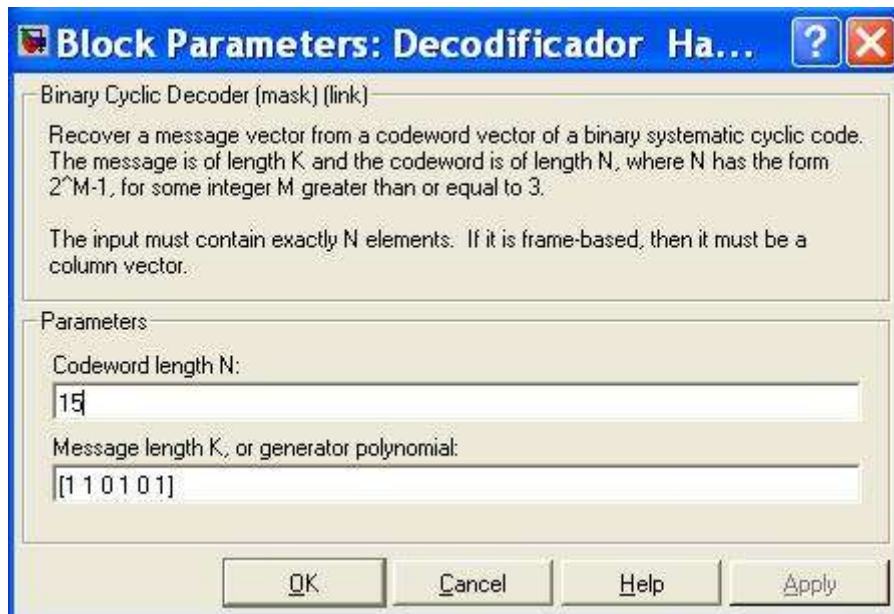


Figura 52. Configuración del decodificador Hamming.

## 5.4 VALIDACIÓN DEL PROGRAMA

Para realizar las pruebas a los módulos se utilizaron visualizadores pertenecientes a Simulink®. Los que muestran las señales resultantes de cada uno de estos módulos, para compararlos con la teoría de Bluetooth.

- Bloque generador de la señal.

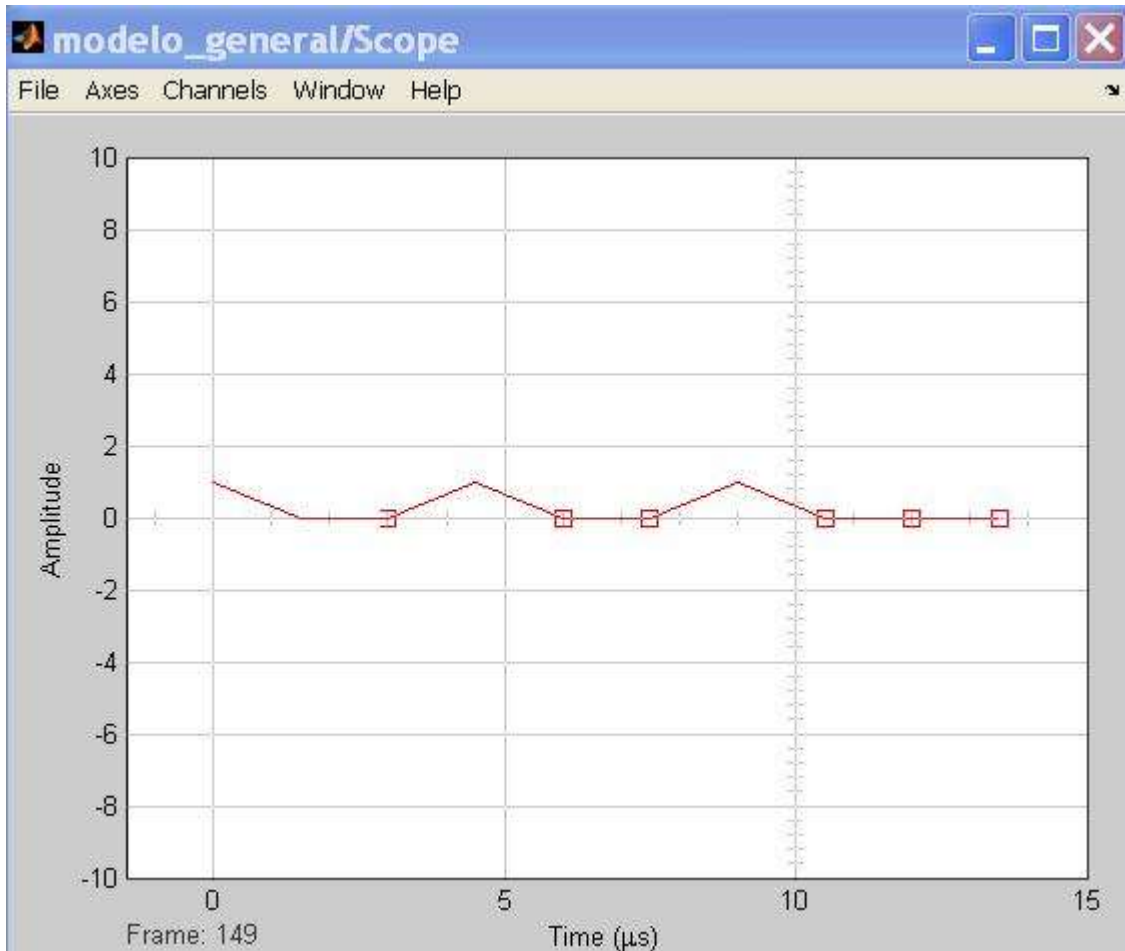


Figura 53. Generador de datos.

En esta figura se observa que el bloque esta generando un flujo de 10 bits con amplitudes ideales de carácter lógico, lo cual está de acuerdo con la teoría.



- Bloque codificador Hamming.

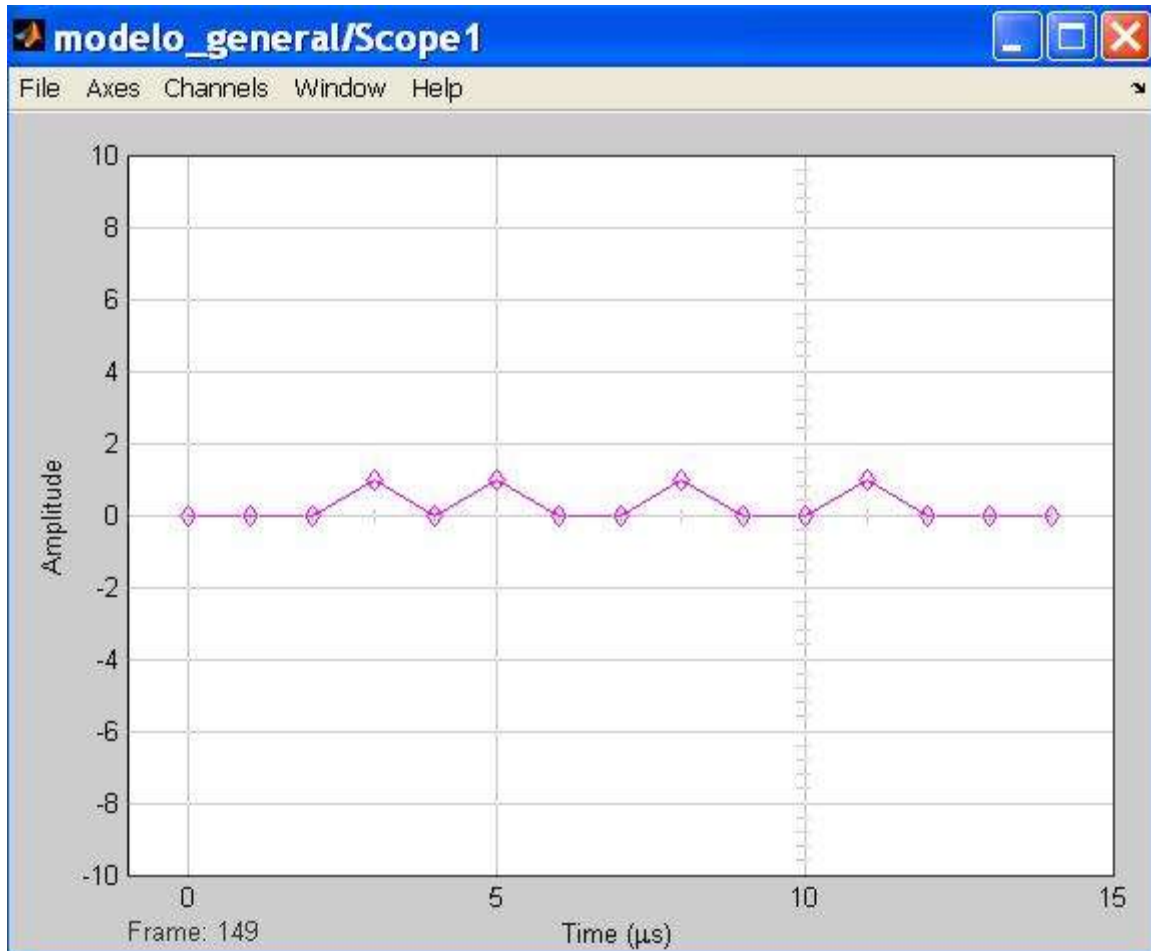


Figura 54. Codificador Hamming.

Este bloque es el encargado de efectuar la operación de codificación mediante el polinomio generador, el resultado esperado para este bloque es una muestra extendida de 15 bits, en el mismo intervalo de tiempo. Con la figura se puede observar que este bloque cumple con los objetivos para los cuales fue creado.

- Bloque de ensamble de paquetes

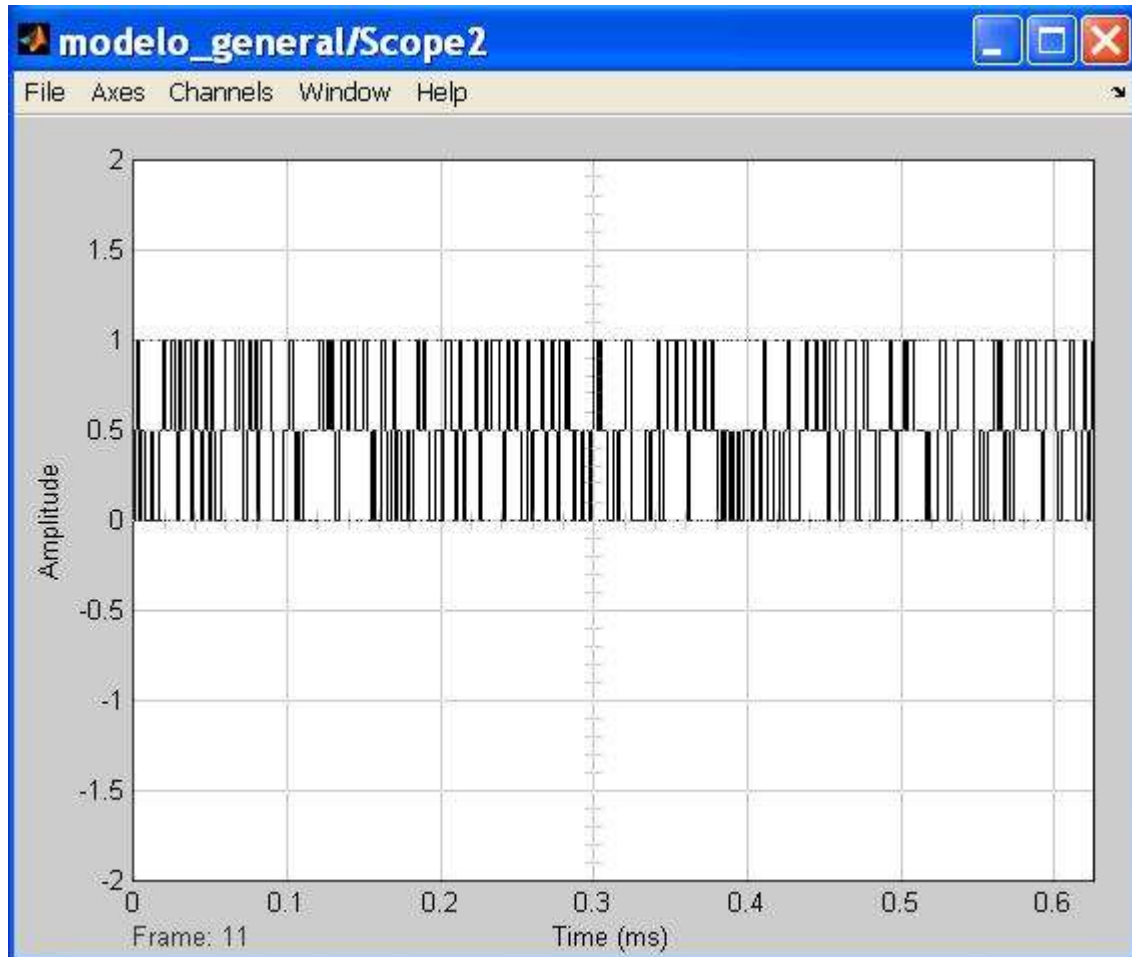


Figura 55. Bloque de ensamble de paquetes.

Se observa la formación de tramas, sin embargo es físicamente imposible el conteo del número de bits de esta, asumiendo por los resultados anteriores y posteriores que este bloque está cumpliendo el objetivo deseado.

- Bloque de modulación GMSK y saltos de frecuencia.

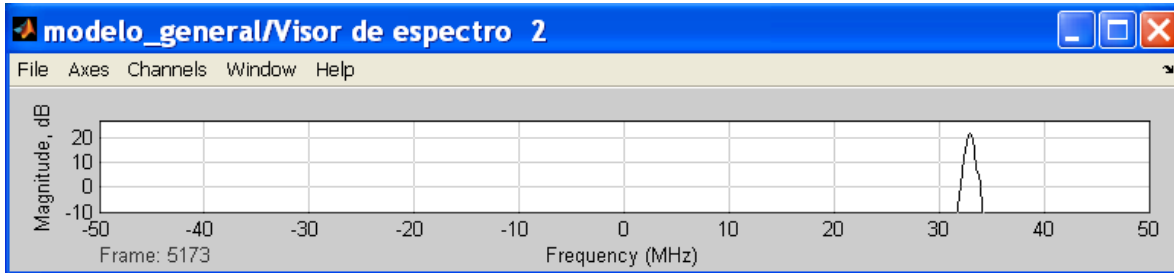


Figura 56. Bloque GMSK y saltos de frecuencia.

En el visor de espectro se ve un desplazamiento de la señal a transmitir, en la simulación se ve el cambio de frecuencia en el tiempo. Por este motivo se ve que el resultado es el adecuado.

- Bloque de canal

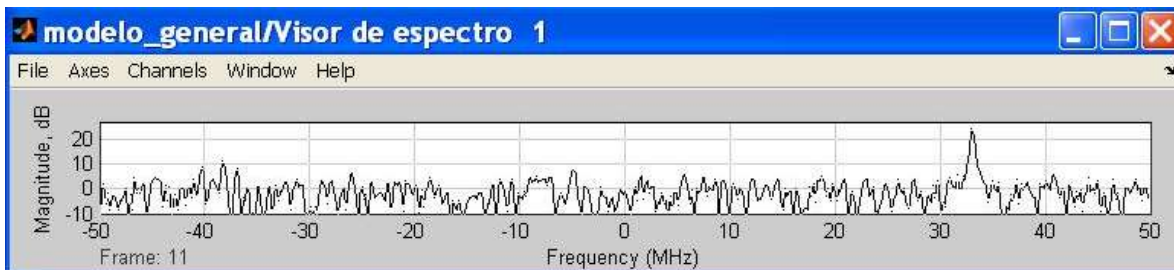


Figura 57. Bloque del canal de TX.

Comparando este visor de espectro con la figura anterior se observa la adición de ruido gaussiano a la señal transmitida, lo que se pretendía con este módulo.

- Bloque generador de los saltos de frecuencia.

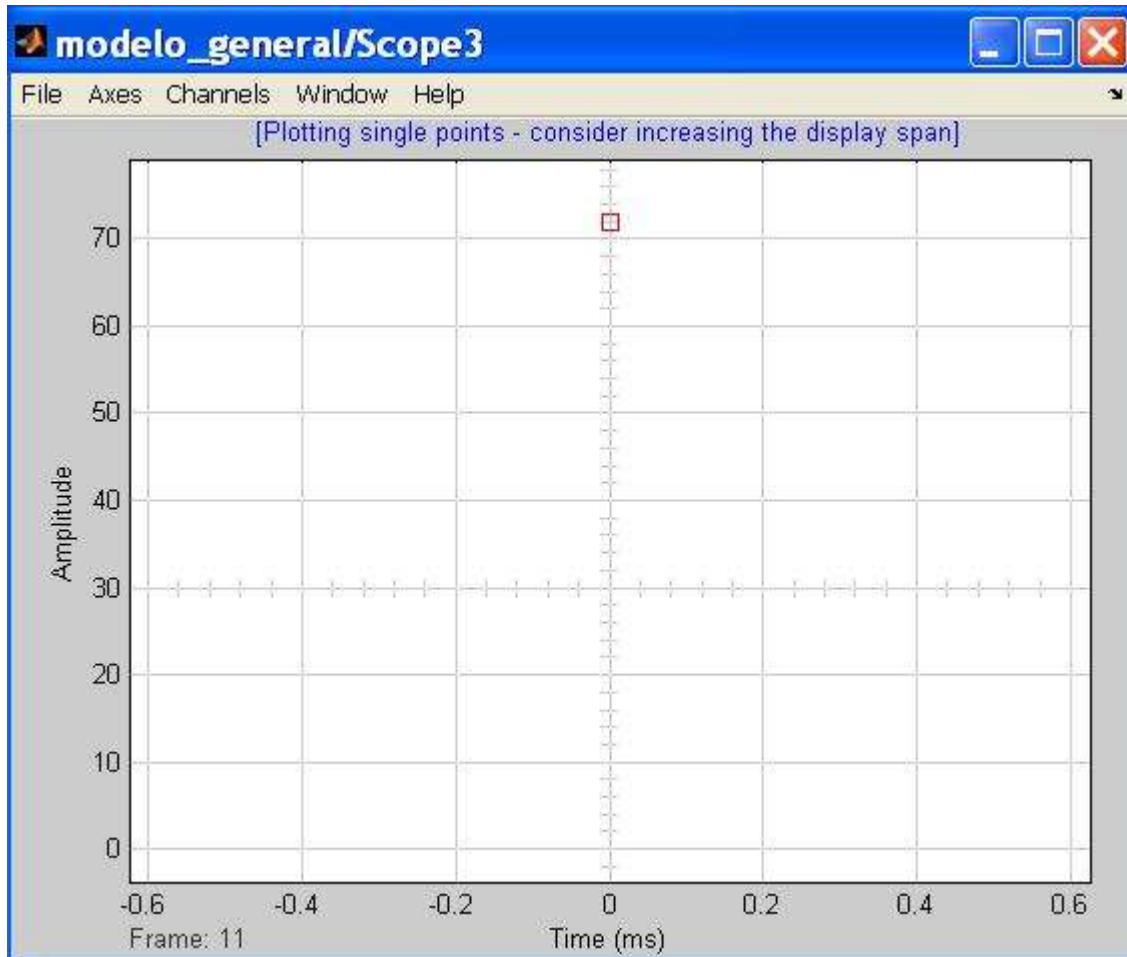


Figura 58. Bloque generador de salto.

En este visualizador observamos que el bloque está generando los valores requeridos entre 0 y 79, con el fin de entregar estos valores escalares al modulador y remodular, con lo que se puede concluir que si cumple los requerimientos para lo cual se creó.

- Bloque demodulador

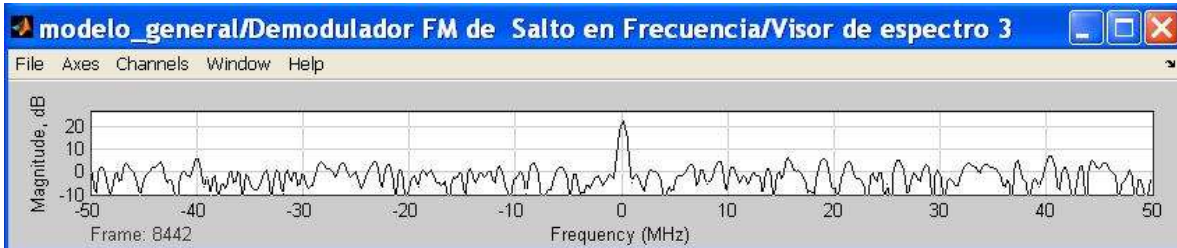


Figura 59. Bloque demodulador.

En el visualizador se observa que este módulo elimina los saltos de frecuencia, generados en el bloque modulador y la deja lista para el desentramado y la decodificación, esto se puede comprobar al observar la simulación.

- Bloque de desensamble de paquetes.

En este visualizador se comprueba que la señal entregada por el demodulador, se reduce a tramas de 15 bits, esto también sirve para comprobar que el bloque de ensamble de paquetes si está cumpliendo su objetivo, porque al aplicar el proceso contrario se obtiene la señal que el había modificado.

Como se mencionó anteriormente este bloque es el encargado de recibir los datos en tramas de 625 bits y transformarlas a tramas de 15 bits, por lo que se puede concluir que este bloque está cumpliendo con los objetivos.

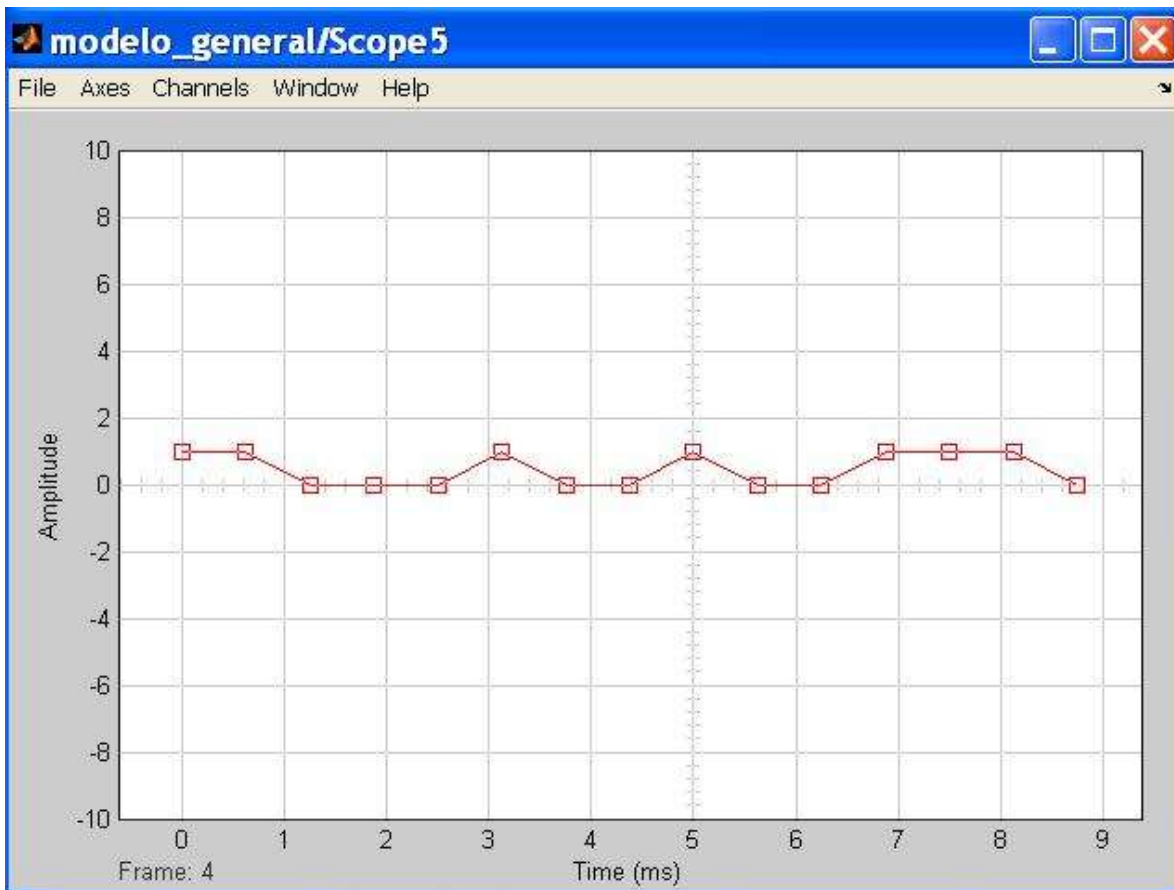


Figura 60. Bloque desensamble de paquetes.

- Bloque decodificador Hamming.

Este bloque es el encargado de retirar el polinomio generador que se adicionó en el codificador y retornar la señal a su estado original, este bloque también se encarga de calcular la tasa de error (BER), la cual se visualiza mediante un despliegue al correr la aplicación.

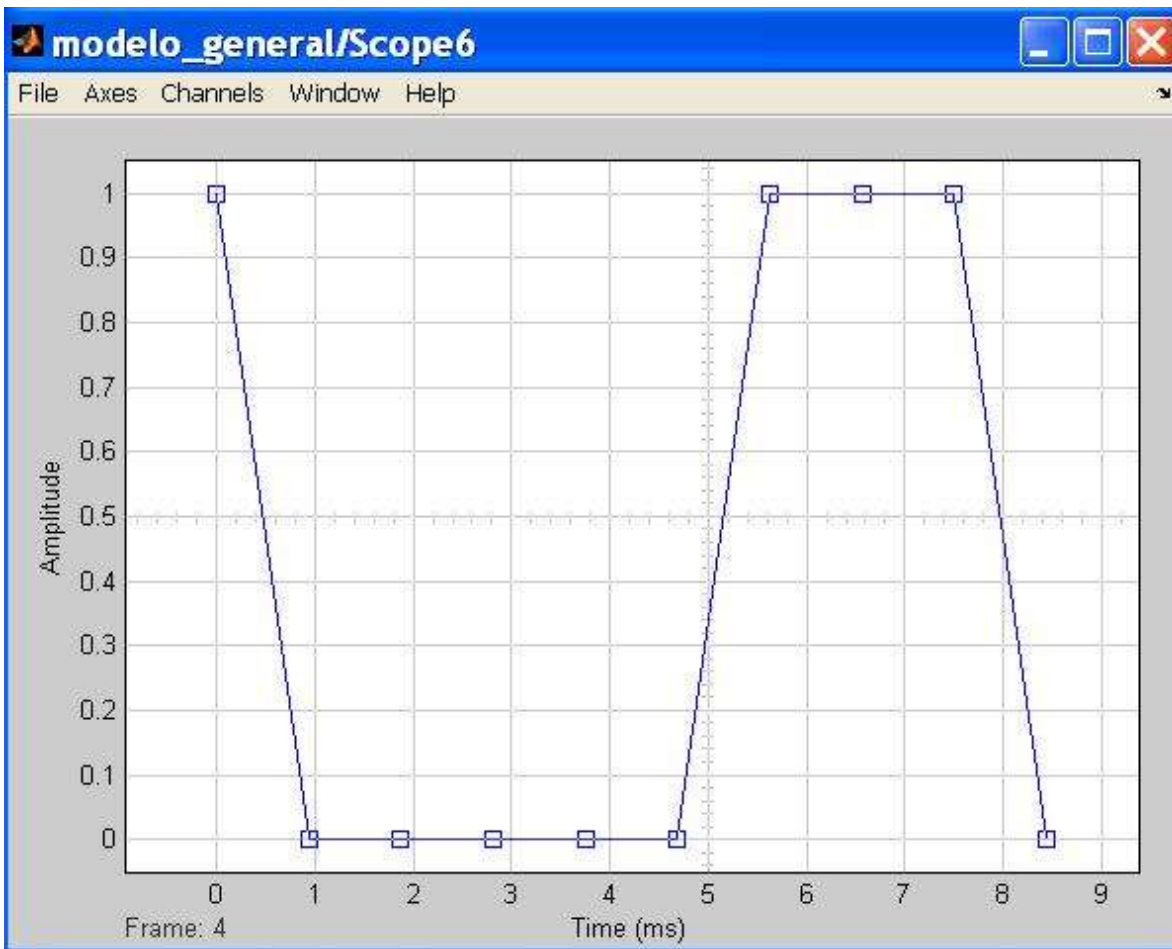


Figura 61. Bloque decodificador Hamming.

Este bloque debe entregar a la salida una muestra de 10 bits, que fue con la que se originó la transmisión, lo cual se comprueba en la gráfica, con lo que se concluye que este bloque cumple con su objetivo.

- **BER**

Es el módulo asociado al cálculo de nivel de error en la recepción de la información que corrobora la funcionalidad de los módulos de codificación y decodificación Hamming. Está conformado por un bloque comparador de señales

bandabase que hace parte de la caja de herramientas de Simulink®, su finalidad es hacer el cálculo de la tasa de error de bits (*Bit Error Rate*, BER) mediante la comparación de los flujos de señal tanto de la información transmitida como de la recibida. La salida del bloque va conectada a una unidad de visualización numérica que representa la interfaz de usuario a través de la cual se puede apreciar la tasa de error.

En la figura se aprecia que se toman datos del transmisor y se compara con los recibidos por el receptor, lo cual se lleva a un despliegue digital.

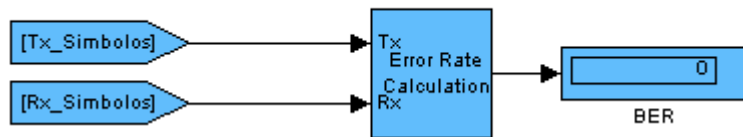


Figura 62. BER

## 5.5 MODELO TOTAL DE LA SIMULACIÓN

Una vez finalizado el análisis hecho a los diferentes bloques, se diagrama la asociación entre estos. Los diferentes bloque han sido diferenciados por grupos de colores.



Este conjunto constituye el sistema total de simulación del nivel físico de Bluetooth.

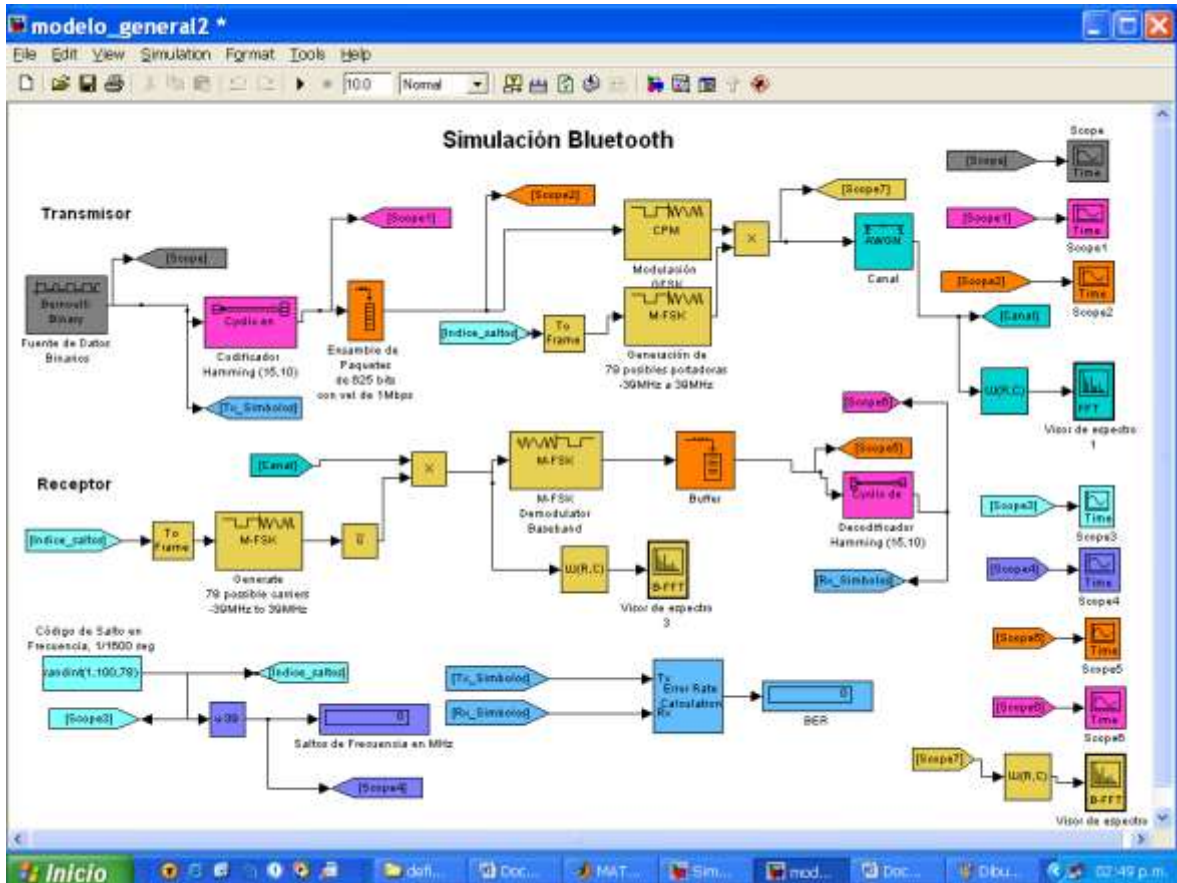


Figura 63. Diagrama total de la simulación.

## 6. CONCLUSIONES Y RECOMENDACIONES

De acuerdo con todo el proceso de documentación y desarrollo del proyecto de grado, se logró obtener una serie de conclusiones y recomendaciones.

Se puede concluir y recomendar:

- El proceso de documentación es una tarea exhaustiva basada en datos confiables, tomados por ejemplo, de sitios oficiales del objeto de investigación, esto permite maximizar la eficiencia en el tiempo de búsqueda y da confiabilidad a la información de respaldo para el documento final.
- Las herramientas empleadas para el desarrollo deben estar suficientemente documentadas. Esto permite tener confianza en el software que soporta la puesta a punto del objeto de investigación.
- Es conveniente tener una visión global y de referencia para diseños reales. Desde este punto de vista, el software de simulación permite alcanzar el objetivo real de una manera más rápida y depurada, impidiendo la pérdida de

tiempo, costos y recursos, contribuyendo al profesional en Telecomunicaciones a efectuar su labor de una manera eficaz.

- Los medios de simulación empleados para el diseño de aplicaciones aproximadas a la realidad deben ser lo mas completos posibles, de ahí que Matlab® y Simulink® sean una alternativa optima por cuanto proporcionan una excelente interfaz gráfica para el modelamiento basado en módulos y un entorno de programación robusto.
- Simulink es una herramienta que permite efectuar un análisis y diseño óptimo de sistemas de comunicaciones, dado que proporciona librerías dedicadas a ello, las cuales se encuentran muy bien documentadas.
- Antes de iniciar el trabajo alrededor de la metodología de diseño, se recomienda verificar las necesidades hardware del software de ingeniería para evitar problemas de ejecución de la aplicación en tiempo de simulación, teniendo en cuenta que herramientas como Matlab y Simulink son muy completas, estas exigen a su vez un sistema de computo adecuado, es decir que la plataforma hardware cumpla con las especificaciones técnicas suficientes para soportar las herramientas de desarrollo.
- Se recomienda que la aplicación de simulación desarrollada en cuestión, se emplee en materias como Sistemas Teoría de Telecomunicaciones I y II, Radiocomunicaciones, Radiocomunicaciones Móviles, Sistemas Inalámbricos

en Comunicación de Datos para el fortalecimiento de los procesos de enseñanza y aprendizaje relacionados con tecnologías inalámbricas, de esta manera se contribuye al proceso de actualización del conocimiento académico y profesional.

- Generalizar el uso de ambientes de simulación para el estudio de nuevas tecnologías, permitiendo con esto mayor entendimiento en los procesos de funcionamiento y/o creación de sistemas de telecomunicaciones.
- El sistema de ensanchamiento de espectro por saltos de frecuencia, presenta muchas ventajas frente al ensanchamiento de espectro por secuencia directa, dando a Bluetooth una ventaja, respecto a otros protocolos, para la transmisión de datos en la banda ISM, la cual es muy utilizada por otros dispositivos dado que es de uso libre.
- Pese a su corto alcance (10 metros) de transmisión, se puede afirmar que Bluetooth, es el sistema indicado para las pico-redes. Esperando que en un futuro muy próximo estos limitantes se superen o desaparezcan totalmente.
- Bluetooth es una tecnología inalámbrica de bajo costo, fácil de implementar en dispositivos para ambientes empresariales y personales mediante elementos electrónicos disponibles en el mercado.

## 7. BIBLIOGRAFÍA Y REFERENCIAS

[1]. Sitio oficial de Bluetooth. [www.bluetooth.com](http://www.bluetooth.com).

[2]. Estándar IEEE 802.15.1. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs) 2002, p. 15-108.

[3]. MULLER, Nathan J. Tecnología Bluetooth. Madrid: McGraw-Hill, 2002, p. 55-60.

[4]. HALSALL, Fred. Comunicación de datos, redes de computadores y sistemas abiertos. 4<sup>o</sup> Edición.

[5]. RODRÍGUEZ Calvachi, Oscar Darío. AYA Coral, Ricardo Andrés. Implementación De Una Red Inalámbrica, BLUETOOTH. Trabajo de grado presentado como requisito para obtener el título de Ingeniero Electrónico, CALI, 2003.

[6]. GONZÁLEZ, Juan José. Conceptos Básicos Y Nuevas Soluciones Bluetooth. Ingeniero de Telecomunicaciones.

[7]. LÓPEZ Ortiz, Francisco. Wireless LAN. El estándar IEEE 802.11

[8]. MUÑOZ, Carlos. MUÑOZ, José. Metodología para la simulación de equipos de telecomunicaciones. Tesis FIET. Popayán, 1995.

[9]. UML. [http://www.tevasoft.com/ADOO2005/pautas\\_OO.pdf](http://www.tevasoft.com/ADOO2005/pautas_OO.pdf)

[10]. ALAMO Cantero, Teodoro. LIMON Marruelo, Daniel. ORTEGA Linares, Manuel. RUIZ Arajal, Manuel. HEREDIA Benot, Guillermo. Introducción al Simulink, Modelamiento y Simulación de Sistemas Dinámicos. Departamento de Ingeniería de Sistemas y Automática, Universidad de Sevilla. 110 p.

Otros sitios recomendados:

<http://es.wikipedia.org/wiki/Bluetooth>.

[http://www.arcelect.com/DSSS\\_FHSS-Spead\\_spectrum.htm](http://www.arcelect.com/DSSS_FHSS-Spead_spectrum.htm)

<http://www.monografias.com/trabajos14/wi-fi/wi-fi.shtml>

<http://www.emc.york.ac.uk/reports/linkpcp/appD.pdf>

<http://ie.fing.edu.uy/ense/assign/celular/Teorico/Monografias/Monografia%20de%20CDPD.pdf>