



UNIVERSIDAD NACIONAL DE CHIMBORAZO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

**“Trabajo de grado previo a la obtención del Título de Ingeniero en Electrónica y
Telecomunicaciones”**

TRABAJO DE GRADUACIÓN

Título del proyecto

**“OPTIMIZACIÓN DE LAS TÉCNICAS DE COMPRESIÓN DE AUDIO PARA
TRANSMITIR SOBRE UNA RED ZIGBEE EN UNA PLATAFORMA DE WSN “**

Autor:

FLABIO YUMISACA

DIRECTOR: ING. GIOVANNY CUZCO

Riobamba – Ecuador

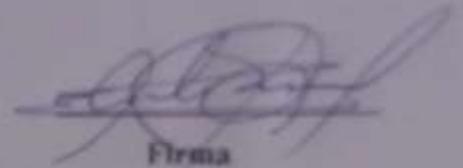
AÑO2018

Los miembros del Tribunal de Graduación del proyecto de investigación de título:
"OPTIMIZACIÓN DE LAS TÉCNICAS DE COMPRESIÓN DE AUDIO PARA
TRANSMITIR SOBRE UNA RED ZIGBEE EN UNA PLATAFORMA DE WSN"
presentado por: Flabio Rodrigo Yumisaca Pinduisaca, y dirigida por: Ing. Giovanny Cuzco.

Una vez escuchada la defensa oral y revisado el informe final del proyecto de investigación
con fines de graduación escrito en la cual se ha constatado el cumplimiento de las
observaciones realizadas, remite la presente para uso y custodia en la biblioteca de la
Facultad de Ingeniería de la UNACH.

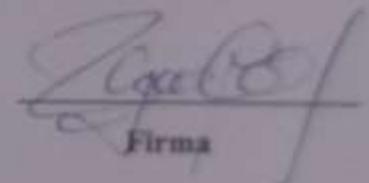
Para constancia de lo expuesto firman:

Ing. Juan Carlos Cepeda
Miembro del Tribunal



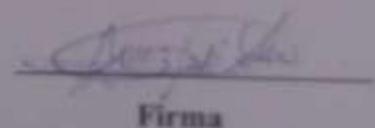
Firma

Ing. Giovanny Cuzco
Director del Proyecto



Firma

Ing. José Jinéz
Miembro del tribunal

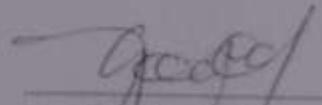


Firma

CERTIFICADO DEL TUTOR.

Certifico que el presente trabajo de investigación previo a la obtención del grado de Ingeniero en ELECTRÓNICA Y TELECOMUNICACIONES. Con el tema "OPTIMIZACIÓN DE LAS TÉCNICAS DE COMPRESIÓN DE AUDIO PARA TRANSMITIR SOBRE UNA RED ZIGBEE EN UNA PLATAFORMA DE WSN" ha sido elaborado por el estudiante Flabio Yumisaca, el mismo que ha sido revisado y analizado en un cien por ciento con el asesoramiento permanente de mi persona en calidad de Tutor por lo que se encuentra apto para su presentación y defensa respectiva

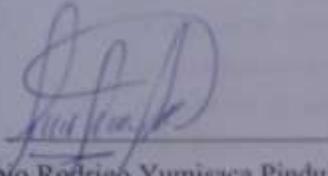
Es cuanto lo que puedo informar en honor de la verdad.



Ing. Giovanni Cuzco

AUTORÍA DE LA INVESTIGACIÓN

La responsabilidad del contenido de este Proyecto de Graduación, nos corresponde exclusivamente a: Flabio Rodrigo Yumisaca Pinduisaca e Ing. Giovanni Cuzco y el patrimonio intelectual de la misma a la Universidad Nacional de Chimborazo



Flabio Rodrigo Yumisaca Pinduisaca

C.I060420401-6

AGRADECIMIENTOS

Doy gracias a Dios por haberme acompañado y guiado a lo largo de mi carrera, en los obstáculos y barreras que en mi vida no me dejaron proseguir y a mis padres que día tras día me ayudaron con sus consejos y fuerzas para salir de mis obstáculos.

Flabio Yumisaca

DEDICATORIA

Dedico este trabajo primeramente Dios, por haberme dado a vida y permitirme el haber llegado hasta este momento tan importante de mi formación profesional. A todos que confiaron en mí y salir adelante.

A mis hermanos por estar siempre presentes, acompañándome para poder realizar mis objetivos.

Flabio Yumisaca

ÍNDICE GENERAL

UNIVERSIDAD NACIONAL DE CHIMBORAZO.....	1
TÍTULO DEL PROYECTO.....	1
1. FUNDAMENTACIÓN TEÓRICA.....	2
1.1. ANTECEDENTES.....	2
1.2. ALGORITMOS DE COMPRESIÓN.....	4
1.2.1. Características del Algoritmo de compresión.....	5
1.3. COMPRESIÓN DE AUDIO.....	6
1.3.1. Algoritmo de compresión sin pérdida.....	6
1.3.2. Algoritmo de compresión con pérdida.....	7
1.4. ALGORITMOS DE COMPRESIÓN DE AUDIO MÁS UTILIZADOS.....	7
1.4.1. Algoritmos de compresión de audio sin pérdida.....	8
1.4.2. Algoritmos de compresión con pérdida.....	12
1.5. DISPOSITIVOS INALÁMBRICOS.....	21
1.5.1. Zigbee.....	22
1.5.2. Xbee.....	31
1.6. RED (Wireless Sensor Networks).....	32
1.6.1. Topología.....	33
1.7. DISPOSITIVO PROGRAMABLE ARDUINO.....	34
1.8. MÓDULO DS3231.....	34
2. METODOLOGÍA.....	36
2.1. TIPO DE ESTUDIO.....	36
2.2. MÉTODOS, TÉCNICAS E INSTRUMENTACIÓN.....	36
2.2.1. Métodos.....	36
2.2.1.1. Analítico/Deductivo.....	36
2.2.2. Técnicas.....	36
2.3. POBLACIÓN.....	37
2.4. MUESTRA.....	37
2.5. HIPÓTESIS.....	37
2.6. OPERACIONALIZACIÓN DE VARIABLES.....	38
2.7. PROCEDIMIENTO.....	39

2.8.	PROCESAMIENTO Y ANALISIS	39
2.8.1.	Parámetros de evaluación de algoritmos de compresión.....	40
2.9.	ANÁLISIS DE LOS ALGORITMOS DE COMPRESIÓN DE AUDIO	40
2.9.1.	Capacidad de compresión.....	41
2.9.2.	Calidad de compresión	43
2.9.3.	Diseño del Circuito Transmisor	44
2.9.4.	Diseño del circuito repetidor	50
2.9.5.	Diseño del circuito receptor	51
2.10.	PROGRAMACIÓN DE DISPOSITIVOS INALÁMBRICOS XBEE.....	57
2.10.1.	Configuración de la red xbee serie 1	57
3.	RESULTADOS.....	60
3.1.	RESULTADOS DE LAS PRUEBAS REALIZADAS EN LA TRANSMISIÓN DE DATOS CON LOS DISPOSITIVOS INALÁMBRICOS.	60
3.2.	PRUEBAS REALIZADAS DE LA POTENCIA DE LOS XBEE S1	61
3.3.	COMPROBACIÓN DE LA HIPÓTESIS	63
3.3.1.	Determinación de los valores estadísticos.....	63
4.	DISCUSIÓN	67
5.	CONCLUSIONES Y RECOMENDACIONES.....	68
5.1.	Conclusiones	68
5.2.	Recomendaciones.....	69
6.	PROPUESTA.....	70
6.1.	Título de la propuesta.....	70
6.2.	Introducción	70
6.3.	Objetivos	70
6.3.1.	Objetivos Generales	71
6.3.2.	Objetivos Específicos.....	71
6.4.	Fundamentación científico – técnica.....	71
6.5.	Descripción de la propuesta	71
6.6.	Diseño Organizacional	72
6.7.	Monitoreo y evaluación del proyecto.....	72
7.	BIBLIOGRAFÍA.....	75
8.	ANEXOS.....	77

ÍNDICE DE FIGURA

Figura 1. Onda del formato wav.....	9
Figura 2. Frecuencias de formato de audio Flac	11
Figura 3. Frecuencia de muestreo de mp3.....	13
Figura 4.Frecuencia de muestreo de AAC	15
Figura 5. Codificador ADPCM	19
Figura 6. Decodificador ADPCM	20
Figura 7. Sensores Inalámbricos	21
Figura 8. Formato de trama zigbee.....	25
Figura 9. Capas del protocolo Zigbee	25
Figura 10. Topología Zigbee.....	28
Figura 11. Aplicaciones de red Zigbee.....	31
Figura 12. Xbee S1.....	32
Figura 13. Red de sensores wsn	33
Figura 14. Arduino Nano	34
Figura 15. Modulo sd-card.....	35
Figura 16. Diagrama de Bloque del sistema.....	40
Figura 17. Compresión de datos.....	41
Figura 18. Compresión de varios Algoritmos	42
Figura 19. Calidad de compresión.....	43
Figura 20. Diagrama de Sistema Transmisor	44
Figura 21. Circuito sujetador.....	45
Figura 22. Arduino Nano y sd-card.....	46
Figura 23. Dispositivos Xbee	46
Figura 24. Circuito electrónico transmisor.....	47
Figura 25. Circuito Transmisor en 3 y en ARES -Software Proteus.....	48
Figura 26. Diagrama de Flujo del Transmisor	49
Figura 27. Diagrama Repetidor.....	50
Figura 28. Arduino nano y el modulo sd-card.....	51
Figura 29. Diagrama del Sistema Receptor.....	51
Figura 30. Arduino nano y cd car.....	52
Figura 31. Filtro Pasa Banda.....	53
Figura 32. Circuito Receptor	54
Figura 33. Diseño de circuito receptor en ARES -Software Proteus	55
Figura 34. Diagrama de flujo del Receptor	56
Figura 35. Configuración del dispositivo Xbee S1	57
Figura 36. Configuración de los modula xbee S1	58
Figura 37. Configuración de los módulos Xbee s1	58
Figura 38. Efectiva de Reproducción.....	61
Figura 38. Toma de Datos dBm	62
Figura 39. Valores obtenidos a varias distancias	62
Figura 40. Diseño Organizacional para el desarrollo del proyecto	72

INDICE DE TABLAS

Tabla 1. Calidad de varios formatos de compresión de audio.....	16
Tabla 2. Operacionalizacion de variables.....	38
Tabla 3. Datos obtenidas de las pruebas	63
Tabla 4. Frecuencias Teórica calculado	64
Tabla 5. Valores del Chi cuadrado	65
Tabla 6. Grados de libertad y probabilidad	65

RESUMEN

En la actualidad existe distintos métodos de comunicarnos entre los más utilizado es la tecnología inalámbrica que facilitan la comunicación a distintas distancias, ya que no necesitan un medio físico, con los dispositivos inalámbricos el usuario tiene la facilidad de transmisión de datos, esto ha conllevado a la utilización de algoritmos de compresión que es utilizados en varios campos de la comunicación por medio de la transmisión de datos en medios inalámbricos, pero para nuestro cometido se analizó algunos algoritmos de compresión de menor pérdida de datos para la voz y audio. El algoritmo de compresión posee un sinnúmero de ventajas como desventajas, y la más importante es que con eso involucra algunas pérdidas de información en todo su proceso. Varios de los algoritmos que se analizaron durante este proyecto han sido adoptadas en la industria de las telecomunicaciones como en la telefonía, internet, imágenes. Pero en el proyecto se utilizó el algoritmo de compresión ADPCM por la características y pruebas realizadas que se determinó que es adecuado para trabajar con dispositivos electrónicos de baja capacidad de memoria y que cumple con el objetivo de enviar datos comprimidos a una distancia moderada, entre los distintos depósitos inalámbricos se tiene el xbee, por su característica que no son complejas y nos ayudan para crear una red confiable y robusta, que para el desarrollo del prototipo fue utilizado el software Arduino y empleado las librerías del Atmega328p, en donde es programado el algoritmo de compresión ADPCM, que los datos comprimidos serán enviados por medio de dispositivos Xbee. El prototipo recibe la información del cable de audio y la información es ingresa al módulo Aduino atmega328, en donde es aplicado el algoritmo de compresión y almacena en un módulo Sd card que será enviado por los dispositivos Xbee y por medio de pruebas realizadas a distintas distancias que nos ayuda en verificar y comprobar el algoritmo de efectividad que se puede implementar en un dispositivo Atmega328 y que en la red de dispositivos xbee la potencia va decayendo de acuerdo a la separación de los dispositivos hasta llegar a la distancia máxima y la potencia de la señal es mínima.

ABSTRACTO

INTRODUCCIÓN

La utilización de algoritmos de compresión es muy importante en las comunicaciones, ya que nos ayuda en la compresión de datos y obtención de una calidad de audio excelente al momento de recibir datos con la menor pérdida posible.

La compresión de datos utilizados en la voz nos ayuda a emplear medios inalámbricos que por su limitado ancho de banda y su insuficiente velocidad de transición se optó por analizar y seleccionar el algoritmo que se adapte mejor para nuestro cometido.

Las tecnologías inalámbricas han presentado un apogeo importante en la actualidad, este desarrollo se debe a las ventajas que se muestran sobre las tecnologías en las cuales resaltan, siendo este modelo la mejor propuesta para la reducción del consumo energético y el costo. En estas características resaltan los dispositivos Xbee por su facilidad de configuración y uno de los inconvenientes de los dispositivos es que se encuentra entre las redes de menor ancho de banda, cuestión por lo cual su uso es usualmente encaminado a la transmisión de señales provenientes de sensores que no requieran altas tasas de envío de datos. Para la compresión de datos en el presente proyecto se utilizó el algoritmo que previamente fue seleccionado después de varias pruebas realizadas en el software Matlab, que cumple con las especificaciones que requiere el dispositivo eléctrico Atmega328 en donde fue programado.

Las características del dispositivo Atmega328 nos obligan a incorporar una tarjeta externa Sd-car, que cumple con el objetivo de almacenar datos que serán transmitidos por la red de los dispositivos xbee, que están conformados por un Coordinador quien transmite los datos hacia el dispositivo Router que funciona como repetidor y los datos son enviados hacia el dispositivo destino final que es acoplado el Atmega328 receptor, que está incorporado una tarjeta externa que almacena los datos y filtrado y a su vez por amplificador para los altavoces.

Con la realización de este proyecto se desea facilitar el análisis del comportamiento de los algoritmos más utilizados en Telecomunicaciones y que se puede enviar datos a distancias moderadas con dispositivos que tienen limitaciones para transmisiones de datos continuos.

CAPÍTULO I

1. FUNDAMENTACIÓN TEÓRICA

1.1. ANTECEDENTES

El presente proyecto de investigación está orientado en la compresión y transmisión de voz por una red inalámbrica, dentro del tema se ha realizado el estudio y análisis de proyectos o innovaciones tecnológicas relacionado al tema o referente a la misma. Por eso se debe obtener un conocimiento previo al desarrollo del proyecto con fines de encontrar soluciones adecuadas en base a la experiencia obtenida de diferentes autores.

En el proyecto de final de carrera en donde menciona que las redes Zigbee como un protocolo de comunicaciones, que es basado en el estándar para redes inalámbricas IEEE_802.15.4. Concebido para el control y la monitorización de redes de sensores tanto en entornos industriales, médicos, como domóticas, ha existido un creciente interés por evaluarlo en aplicaciones del sonido. Aun sin garantizar QoS (Quality of administration) por su limitado ancho de banda, existen un conjunto de aplicaciones para vigilancia, grupos de rescate y salvamento y seguridad, donde los grupos desplegados en un área limitada con necesidad de comunicación utilizan un sistema de audio de bajo costo basado en tecnología Zigbee, que es una comunicación sumamente atractiva. Se presenta el diseño de un sistema que permita la comunicación de un grupo de usuarios desplegadas en un área limitada, empleando los dispositivos RISC, y tecnología zigbee. Se investiga la factibilidad de usar la tecnología inalámbrica para la transmisión de datos, se analizan varias técnicas de compresión de voz para lograr un sinnúmero de usuarios compartiendo el limitado ancho de banda. Las redes Zigbee son utilizadas para el envío de datos voz con un porcentaje mínimo de bit rate, es un estándar que no está pensado para este tipo de aplicaciones pero en determinadas situaciones donde no exista la necesidad de un dispositivo transmisor de alta calidad puede ser utilizado los dispositivos de bajo costo. (Delgado León, 2012)

En el final de carrera, se señal el proyecto de un sistema electrónico que utiliza módulos Xbee para transferir señal de voz. Dichos módulos operan sobre el estándar inalámbrico

802.15.4, el cual no está creado originalmente para transmitir señal de audio. Se presenta una maqueta con los dispositivos inalámbricos configurados para una red, y también se presenta el circuito impreso los módulos que da apoyo a los dispositivos inalámbricos configurados para la red, así como su funcionamiento. Utilizan un módulo para la conversión A/D para adquirir la señal de voz, y la digitaliza, la comprime y la empaqueta, para luego ser guiado por un protocolo esquematizado específicamente para tal efecto. Los paquetes son entregados vía serie a un módulo Xbee que permite la comunicación inalámbrica con otro módulo Xbee situado en una distancia moderada, el dispositivo Router del coordinador. El módulo Router realiza la función de receptor de datos logrando restaurar la señal de voz. Además, se muestra la configuración de los diferentes tipos de programación que conforman el protocolo de comunicación, que son diseñados para que el coordinador controle al módulo terminal. Finalmente, se exponen los resultados obtenidos al someter a varios tipos de pruebas de funcionamiento. (Castellanos Rodríguez, Padrón Quindemil, & Martínez Suárez, 2015)

El trabajo de fin de carrera donde se realizó las comparaciones de los tipos de algoritmos que se utiliza en compresión de audio para transmitir sobre una red con dispositivos inalámbricos Xbee, en donde los dispositivos que forman parte de una red, son más eficientes si son configurados en el mismo canal y a la misma velocidad de transmisión, y para reducir el nivel de los datos se utiliza algoritmos de compresión con pérdida, en donde realizan pruebas de los algoritmos más utilizados y resaltar el mejor para su cometido que cumpla, con comprimir datos con la menor pérdida de información, y determinar que algoritmo es el más efectivo al realizar una transmisión de audio en dispositivos xbee. (LÓPEZ RODRÍGUEZ, 2014)

Hay muchas aplicaciones que requieren establecer una red de bajo costo con una baja tasa de transmisión. Estas redes se denominan LR-WPAN. (Low Rate Wireless Personal Area Network). Hay diferentes soluciones para este tipo de redes, no obstante, su costo es elevado, y son orientadas a un problema en particular e incompatibles entre dispositivos con diferente protocolo. El lenguaje de comunicación es muy robusto, y es un estándar para las LR-WPAN que provee una solución simple y de bajo costo en la programación. Para entornos de con esta aplicación, entre otras, el manejo de redes con sensores y activación de

protocolos de comunicación en domótica, monitoreo ambiental, industria, hospitales y hoteles. En el protocolo de WPAN se definen solo las 2 primeras capas ISO, OSI (la capa física y la de enlace). Donde es especialmente definido para estandarizar las redes de sensores WSN (wireless sensor network). (Dignani, 2015)

El presente proyecto tiene un referente técnico para el desarrollo de temas orientados al procesamiento digital de señales y especialmente a la compresión de audio, ya que, en la actualidad casi todos los sistemas emplean formatos de audio comprimidos, sin pérdidas significativas en la calidad.

En el documento titulado Algoritmos de Compresión, que menciona la utilidad y las técnicas desarrolladas para comprimir, de las cuales, se detallan la compresión con y sin pérdida de información. Dentro de la compresión con pérdida de información se presenta la codificación por transformada, mientras tanto, en la compresión sin pérdida de información se presenta la codificación estadística con la cual la codificación no tiene pérdidas en todo el proceso de codificación de datos, donde toma como algoritmo de compresión la que ocupa filtros de sub-bandas para la reconstrucción de los datos comprimidos del audio. (Pilataxi Duque, 2014)

1.2. ALGORITMOS DE COMPRESIÓN

La compresión es la técnica en donde se reduce el tamaño físico del bloque de datos información, en bloques de datos más pequeños. La compresión de datos se basa en la transformación del volumen de información manejable (procesar, transmitir o grabar).

Que aplicando la compresión esto reduce la tasa de bits, En esencia, compresión significa que la misma información se transfiere usando una menor tasa de datos. Se debe indicar que en el lenguaje habitual del audio, "compresión" significa el procedimiento en el cual se disminuye el rango dinámico de la señal representativa del sonido.

La compresión ejecutada directamente sobre los datos, por ende, es cuestión de almacenar los datos repetidos de un patrón de bits a otro. En otro plano, se lleva a cabo por sustituir esta información por otra similar.

Los algoritmos que utilizan tramas de distinta longitud de bits y cada uno con diferentes caracteres de un fichero. Si se asignan códigos más cortos a los caracteres que aparecen más a seguido se consigue una compresión del fichero más efectivo. Esta compresión de caracteres es mayor cuando menor es su aparición de datos similares.

1.2.1. Características del Algoritmo de compresión

Como compresión se tiene que el espacio que ocupa una información codificada (datos, señal digital, etc.) sin compresión es la relación entre la frecuencia de muestreo y la Fs. Por ende, cuantos más datos se empleen mayor será la capacidad del archivo. No obstante, la decisión viene impuesta por tal motivo, se empleará la compresión, para transmitir la misma cantidad de información que ocuparía un número mínimo de bits.

La información puede ser transmita de tres tipos:

- **Redundante:** Información repetitiva o predecible.
- **Irrelevante:** Que no se puede estimar, cuya eliminación por tanto no afecta al contenido del mensaje.
- **Básica:** La que no es ni redundante ni irrelevante. La que debe ser transmitida para que se pueda reconstruir la señal.

Teniendo en cuenta estos tipos de información, se establecen tres topologías de compresión de la información:

- **Sin pérdidas reales:** Es decir, transfiriendo la información básica e irrelevante, pero eliminando la redundante.
- **Subjetivamente sin pérdidas:** Asimismo de eliminar los datos repetitiva se elimina también la repetitiva.

- **Subjetivamente con pérdidas:** Se descarta cierta cantidad de información esencial, por lo que el mensaje se reconstruirá con errores evidentes pero tolerables.

El objetivo principal de la compresión es continuamente disminuir el tamaño de la información, intentando que esta disminución, no afecte al contenido. También la disminución de datos puede afectar al tamaño y no mucho a la calidad de la información.

1.3. COMPRESIÓN DE AUDIO

La compresión de audio es un aspecto de reducir el tamaño la información única, especificando en la reducción del tamaño de los datos del archivo de audio. Los algoritmos de compresión de audio son, habitualmente llamados códecs de compresión. Existen distintos tipos de algoritmo de compresión, basados en compresión de datos sin pérdida y algoritmo de compresión de datos con pérdida.

La compresión es esencialmente una reducción automática de volumen que hace el compresor, en esencia es reducir el nivel de una señal con unos parámetros que son fijados por el usuario y que modifican a un estado más reducido. Y igualmente disminuye la tasa de bits de una señal digital de audio buscando como fin la reducción de su peso en tamaño. El audio más cercano a una fidelidad idónea, con respecto al sonido real es el normalmente conocido códec WAV. A medida que se comprime el audio se reducen estos valores, con lo cual se pierde calidad con respecto al sonido original. (Huidobro, 2014)

En las cuales consta dos tipos de algoritmos de compresión:

- Algoritmo de compresión sin pérdida
- Algoritmo de compresión con pérdida

1.3.1. Algoritmo de compresión sin pérdida

Los algoritmos de compresión sin pérdida es el procedimiento de codificación, que tenga como objetivo representar tanta cantidad de información utilizando y reducido a un espacio menor, siendo posible una reconstrucción exacta de todos los datos utilizados. Es decir, la

compresión sin pérdidas incluye a aquellas técnicas que garanticen crear un duplicado exacto de la cantidad de datos de entrada después de un ciclo de compresión. Por esta razón es utilizada para comprimir archivos que contienen datos que no pueden ser destituidos o perdidos, como pueden ser documentos de texto, audio e imagen.

1.3.2. Algoritmo de compresión con pérdida

El algoritmo de compresión con pérdida es el método de codificar, que tiene como finalidad la representación de una porción de información, pero utilizando cantidad mínima de la misma, no consiguiendo reconstruir de forma exacta los datos.

La compresión con pérdida solamente es apropiada si la reconstrucción exacta no es útil, para que la información tenga sentido. La información reconstruida es solo una aproximación de la información original.

Suele limitar a la información analógica que ha sido procesada en (imagen, audio, video), donde la información puede ser parecida, y al mismo tiempo, ser subjetivamente la misma. Su mayor ventaja reside en su alta tasa de compresión que ofrece en comparación a un algoritmo de compresión sin pérdida.

En la compresión con pérdidas hay dos técnicas muy comunes:

- **Códec de transformación:** Los datos originales son transformados de tal forma que se simplifican (sin posibilidad de regreso a los datos originales). Creando un conjunto de datos nuevos dado a altas razones de compresión sin pérdida.
- **Códec predictivos:** Los datos originales son analizados para pronosticar el comportamiento de los mismos. Después se compara esta predicción con los datos reales, codificando el error y la información necesaria para la reconstrucción. Nuevamente, el error es propenso a altas razones de compresión sin pérdida.

1.4. ALGORITMOS DE COMPRESIÓN DE AUDIO MÁS UTILIZADOS

En las comunicaciones existe una gran cantidad de formatos de compresión de voz y audio. Los más frecuentes en compresión de audio de mayor calidad es el MP3 (MPEG-2 Audio Layer III). Por lo general, el tipo de formato se corresponde con la extensión del archivo (las iniciales del nombre del archivo después del punto son las extensiones del algoritmo utilizado, por ejemplo .mp3, .wav, .ogg, .wma).

Un códec es igual que algoritmo de codificación y compresión de los datos en un formato de audio. Algunos tipos de archivo tienen asignado un tipo de algoritmo específico. Por ejemplo, el códec MP3 siempre utiliza el formato MPEG Layer-3, mientras que el formato MP4 puede utilizar una gama de códec distinto.

Especialmente cuando un formato siempre usa un algoritmo único. En términos sencillos, un formato se puede comparar con un recipiente en el que se puede almacenar un sonido o audio de una señal de vídeo que utiliza un códec determinado.

1.4.1. Algoritmos de compresión de audio sin pérdida

Este conjunto de formatos sin pérdida, comprime y descomprime un sonido de tal manera que permite la preservación de su calidad original exacta cuando se decodifica.

Los tipos de formatos de algoritmo sin pérdida más comunes son:

- WAV (Waveform Audio Format).
- FLAC (Códec de audio, sin pérdida y libre).
- ALAC (Códec de audio, sin pérdida de Apple).

1.4.1.1. WAV

El algoritmo Wave uno de los iniciales en la transformación de audio. Se utiliza principalmente para codificar pistas de audio sin pérdida de datos (PCM) que son idénticas a los CD de audio en términos de calidad. En promedio, un minuto de sonido con formato

WAV requiere alrededor de 10 Mbps de memoria. Los CD generalmente se digitalizan en formato WAV y luego se pueden comprimir en MP3 con un algoritmo de audio, esto hace que el modo sin pérdida sea ideal para archivar material de audio o cualquier otra situación en la que la calidad sea primordial.

- **Detalles Técnicos sobre el formato wav**

Los archivos de audio WAV son en realidad archivos de audio en bruto, que no están comprimidos de ninguna manera. Por lo tanto, en la mayoría de los casos son significativamente más grandes de lo que se está acostumbrado a ver en otros formatos, como MP3. El motivo de la diferencia se debe a que muchos otros formatos de audio aplican compresión, lo que reduce significativamente el tamaño del archivo.

Para el formato Wav los datos se guardan en porciones de fragmentos, que contienen una etiqueta de cuatro caracteres y el número de bytes de cada porción, el formato WAV puede incluir patrones de 8 a 16 bits con frecuencias de entre 1.1 Khz y 4.41Khz. La mayor calidad aceptable, 16 bits a 44.100 Hz, es la perteneciente al CD, con un almacenamiento de 88 kB/s. La codificación del movimiento de bits normalmente utiliza la forma de modulación por impulsos lineal (LPCM). Los sonidos del sistema de Windows, se almacenan en formato WAV sin comprimir. Como se muestra en la figura 1.

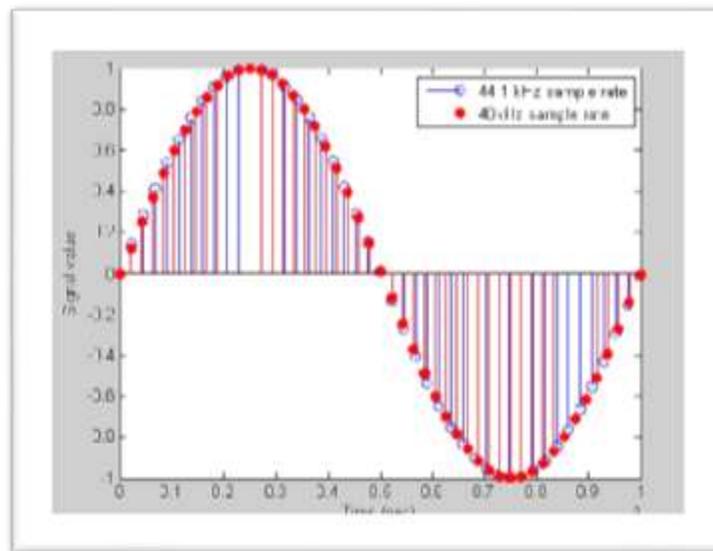


Figura 1. Onda del formato wav
Fuente: <http://blogs.heraldo.es/ciencia/?p=2814>

1.4.1.2. FLAC

FLAC (Free Lossless Audio Codec), un formato de audio semejante al MP3, pero sin pérdida, lo que significa que el audio se comprime en FLAC sin pérdida de calidad. Esto es semejante a cómo desempeña el compresor Zip, a excepción que con FLAC obtendrás una compresión mucho mejor porque está diseñado específicamente para comprimir datos de audio, y puede ser reproducido en tu reproductor favorito, si tiene el códec lo reproduce como si fuera un archivo MP3. La desventaja del algoritmo es que el archivo transformado al formato flac ocupa más espacio del que se obtendría al aplicar un algoritmo de compresión con pérdida. Es una enorme desventaja, de este tipo de formato de compresión con respecto al habitual MP3 o los desarrollados en tiendas que venden canciones para la descarga en formato AAC o WMA, consiste en que logran comprimir el tamaño del archivo sin perder totalmente la calidad, como si se tratara de un compresor ZIP o un RAR, principalmente ideado para la música.

- **Detalles técnicos del formato Flac**

El formato FLAC suele tener la misma extensión como su nombre, aunque no es extraño encontrarlos como son perfectamente reproducibles con gran cantidad de lectores de música, principalmente de una variedad media y alta, incluso en computadoras tradicionales se les puede reproducir, ya que una de las características del algoritmo flac, es que los archivos se descodifican en forma fácil. Estos formatos son de velocidad de bits inestables, que no todas las partes de un mismo formato de audio son igualmente comprimibles.

Otra peculiaridad es que, como todos los algoritmos de compresión sin pérdida, la compresión final depende mucho del estilo musical y la variedad sonora, la existencia de ruido típica de instrumentos como los instrumentos acústicos o la batería importa información al espectro sonoro en casi todas las frecuencias del audio. Para música sin compresión o limpia espectralmente mencionando, con la variedad de compresión para instrumentos acústicos y voz, se obtiene reducciones de aproximadamente 1 a 2 en otros estilos elevados, como el dance o el rock, con un espectro sonoro generalmente más nítida, los archivos pueden quedar desde 2 hasta 4 del tamaño original en formato PCM. Otra causa muy importante es la subsistencia de los silencios, ya que varios audios tienen

segundos de silencio al principio o al final, en el caso de muchas piezas de música artística hay silencios en diversas partes que el algoritmo la comprime y además se puede observar en la figura 2, el espectro de audio de cd con el del algoritmo flac. (Pacheco, 2015)

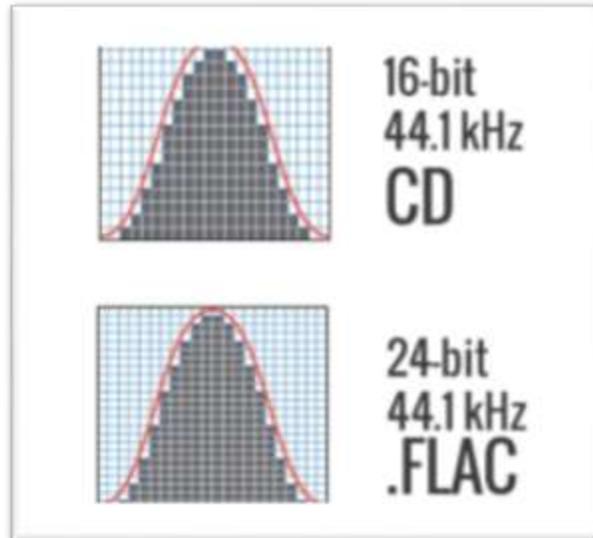


Figura 2. Frecuencias de formato de audio Flac
Fuente: <http://hometech.com.mx/flac-el-estandar-que-viene/>

1.4.1.3. ALAC

El formato ALAC (Apple Lossless Audio Codec - Códec de audio, sin pérdida de la compañía Apple). Es un gran algoritmo para mantener sus canciones sin compresión, pero la gran diferencia es que será idéntico a la fuente original las pistas musicales aplicados este algoritmo. Este formato de audio sin pérdida es similar a otros de los que quizás haya escuchado en donde el audio se codifica utilizando un algoritmo sin pérdidas que se almacena en un contenedor MP4. Incidentalmente, los archivos ALAC tienen la misma extensión de archivo .m4a que AAC, por lo que esta convención de nombres puede generar confusión.

- **Detalles técnicos del formato Alac**

El principal detalle es que mantendrá su calidad de audio por lo alto, puede saber que la conversión de un formato con pérdida a otro degrada la calidad del audio. Sin embargo, si

usa un formato sin pérdida como ALAC, puede convertir cualquier cosa sin perder información de audio.

1.4.2. ALGORITMOS DE COMPRESIÓN CON PÉRDIDA

Cuando el dato se comprime con pérdida, un sonido se somete a cierta modificación. La compresión elimina las frecuencias de sonido que son inaudibles para el oído humano. Cuando el dato se decodifica, el archivo será distinto del original en términos de la información almacenada en el mismo, pero suena prácticamente igual.

Algunos de los formatos con pérdida más comunes son:

- Mp3
- AAC
- OGG
- WMA
- ADPCM

1.4.2.1. Mp3

Es un formato de compresión de datos digital, tipo comprimido con pérdida de información, el formato mp3 es apreciado por el sentido auditivo humano, por tanto, no distinguimos entre un audio sin compresión y un formato de compresión mp3. Igualmente, un archivo de formato Mp3 consigue disminuir el tamaño del archivo del formato sin intervenir en su calidad, un audio en formato mp3 ocupa menor tamaño que un audio formato WAV, con una calidad igual a la calidad de cd de audio normal.

Con la siguiente ventaja el algoritmo de compresión mp3, pueda ser procesado en aproximadamente todos los reproductores de audio, una de las preferibles opciones en estos casos para almacenar música con buena calidad en dispositivo, igualmente, es el formato de audio que más se emplea en reproductores portátiles y de escritorio, el formato mp3 es un estándar y por tanto la afinidad con todos los modos de reproducción está garantizada. El formato mp3 permite elegir la calidad de compresión que se aplicará al audio, y la calidad de cd sería equivalente a 128 Kbps, pero se puede seleccionar la compresión dentro los 8

Kbps y los 320 Kbps de calidad, tomando en cuenta que cuanto más superior sea los datos (Kbps) comprimidos, mayor espacio ocupará el archivo que sea aplicado la compresión mp3.

La velocidad de compresión del mp3 se encuentra entre los rangos de 1.6 KHz y los 48 KHz, y tan solo mantiene 2 canales (estéreo).

- **Detalles Técnicas del formato mp3.**

El apelativo técnico para el formato mp3 es ISO MPEG Audio Layer 3. De este renombre concluimos que pertenece al grupo de los MPEG de audio 3. El algoritmo de compresión es con pérdida de datos. El sentido del oído humano no escucha todas las frecuencias de audio que un artefacto musical reproduce, especialmente las frecuencias que se encuentran en los límites de la apreciación humana. Una de los métodos consiste en deshacer esas pequeñas porciones de archivos de música que apenas son audibles por el sentido del oído humano. También descarta sonidos bajos que se opacan con sonidos fuertes, pues tampoco se puede escucharlos.

Lo más usual que el formato MP3 se configuren en 44.1 KHz canal estéreo y entre 64 y 128 kbps. Finalmente, la calidad del audio de un formato MP3 es muy similar al de un CD. (Rodríguez Navarro, 2012), como se muestra en la figura 3.

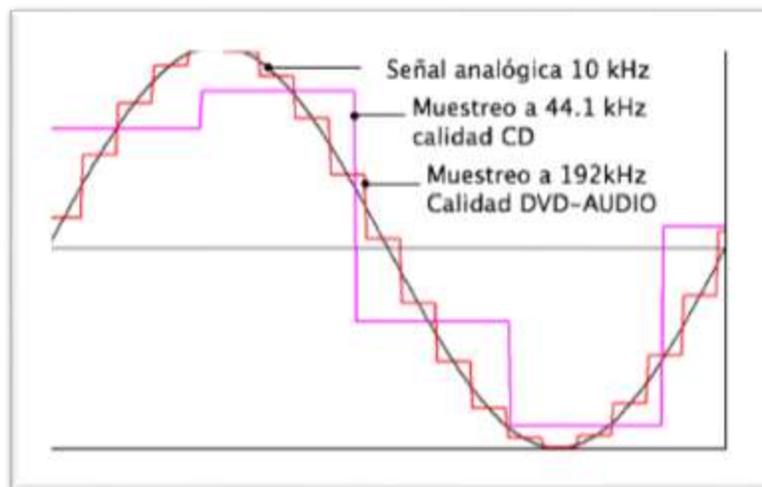


Figura 3. Frecuencia de muestreo de mp3
Fuente: <http://highresolutionaudio.blogspot.com/>

1.4.2.2. AAC

El algoritmo AAC es un modelo de compresión de audio digital como extensión de MPEG-2, es un formato de compresión con pérdidas de datos, y promete más calidad que el mp3, y es más fijo para una misma velocidad de transmisión que son de tamaños iguales. Su codificación está basada con los mismos inicios del formato MP3, con la semejanza de que ofrece el empleo de frecuencias de muestreo de 8 Hz hasta los 96 KHz. El procedimiento de codificación, transforma automáticamente el número de Bitrate necesarios en función de la dificultad de la transmisión de datos en cada momento.

El formato de compresión AAC soporta hasta 48 canales como último, lo que ofrece y sea el formato especial en sonido envolvente, es decir es una buena opción en caso de no escuchar el audio en un dispositivo reproductor, y en el formato de vídeo. Es más eficaz que el formato MP3 en casi todas las apariencias, ofrece una buena calidad y archivos de menor tamaño, pero no es tan popular y compatible con el formato Mp3.

- **Detalles Técnicas del formato AAC**

El formato utiliza una compilación variable de bits (VBR), un procedimiento que adapta el número de bits empleados para reducir datos de audio, según la dificultad de trasladar datos en un momento definido. El formato que se utiliza tiene una utilidad superior al MP3, que realiza una mejor calidad en datos más pequeños y requiere mínimos recursos del sistema para comprimir y descomprimir.

Los dos canales estéreo como máximo que el formato mp3 dispone, el formato AAC dispone de 48 canales polifónicos máximos, y ofrece frecuencias desde 8 Hz a 96 KHz, en diferencia al formato mp3.

Con el formato AAC los archivos se guardan con la extensión Mp4 (MPEG-4), los archivos tienen menor tamaños que los de formato Mp3. (Rodríguez Navarro, 2012), como se observa en la figura 4.

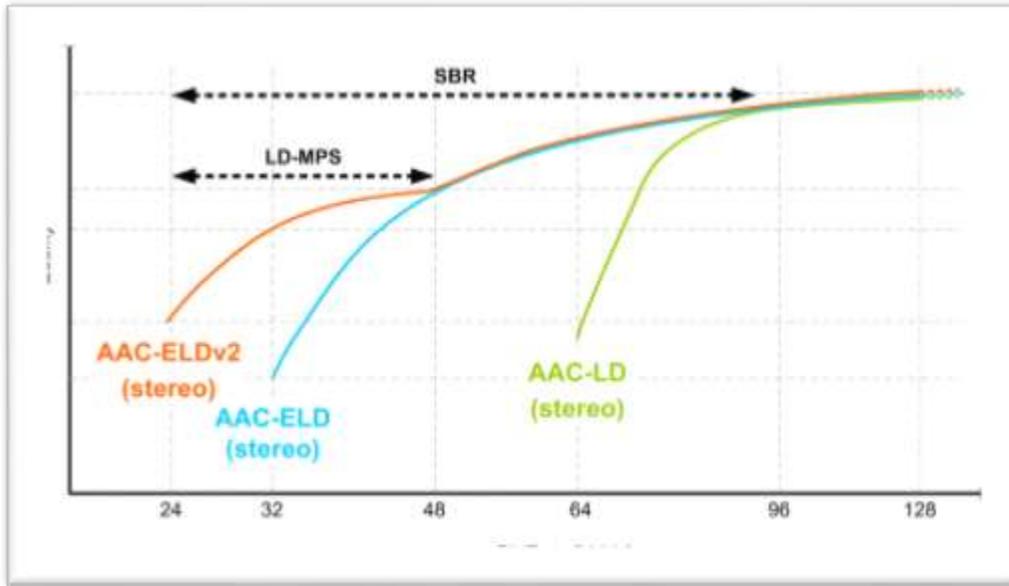


Figura 4. Frecuencia de muestreo de AAC

Fuente: <https://www.iis.fraunhofer.de/en/ff/amm/prod/kommunikation/komm/aaceld.html>

1.4.2.3. OGG

Es un formato de compresión de audio digital con pérdida de datos, normalmente los datos de formato Ogg están codificados con el formato Vorbis, que es un algoritmo independiente que permite una alta flexibilidad al momento de elegir entre los diferentes tipos de velocidad de transmisión (bitrate) según la dificultad de la transmisión de datos, en el nexo de calidad, se encuentra al igual con MPEG-2 y en la mayoría de formatos tienen una velocidad de transmisión estable, es comparable al formato ACC.

El códec ogg está seleccionado para modificar desde el nivel de calidad del teléfono que es esta muestreado a 8 kHz inclusive a su máximo nivel de 192 KHz, y para sistemas estereo, armónicos, cuadrafónicos, 5.1, ambiosonidos y hasta 255 canales discretos. Los bitrat es disponibles van desde 32 Kbps hasta 500 Kbps. El códec Ogg dispone de una mejor calidad de audio entre 8 KHz y 48 KHz que el formato mp3 y sus datos ocupan menos espacio. El uso con dispositivos y softwares no son muy compatibles, tampoco es un formato todavía tan universal como el formato mp3, pero cada vez más dispositivos y programas lo reconocen y pueden trabajar con él formato con mayor estabilidad.

- **Características Técnicas del formato Ogg.**

El formato Ogg tiene una codificación variable de la frecuencia de bits (VBR), un procedimiento de compresión que varía el número de bits utilizados para modificar datos de audio, en función de la dificultad de la transmisión de audio en un instante determinado. Es decir, que la cantidad de compresión de audio que se utiliza para codificar una cantidad determinada de datos no es lo mismo para el silencio que para un audio de instrumentos tocados.

Mientras que el MP3, sólo puede grabar un límite máximo de dos canales, el formato Ogg puede grabar polifónicamente (varios canales) y, por lo tanto, puede reproducir audios en procesos que utilizan canales 4.5 o 7 (7.1).

Como el códec Ogg Vorbis es más nuevo que el MP3 (la versión 1.0 de Ogg inicio en el mercado el año 2000), ofrece una mejor fidelidad de audio, en una escala de frecuencia.

Los archivos que se encuentran en la tabla 1, se pueden utilizar para comparar la calidad de varios formatos de compresión de audio, en velocidad vinaria media y baja y comprar que formato es adecuado en esas velocidades. (Rodriguez Navarro, 2012)

Tabla 1. Calidad de varios formatos de compresión de audio.

Recopilación	Ogg vorbis	WMA	MP3	MP3 Pro
Velocidad vinaria media	.ogg .wav	.wma .wav	.mp3 .wav	
Velocidad vinaria baja	.ogg .wav	.wma .wav	.mp3 .wav	.mp3 .wav

Fuente: Autor

1.4.2.4. WMA

El formato de Audio (WMA) es una tecnología de compresión con pérdida y es desarrollada por la empresa Microsoft. El nombre puede usarse para referirse al formato de archivo de audio o al códec de audio (WMA). Es el software propietario del formato es

el Windows Media. El formato consiste de cuatro códecs distintos, el códec WMA original, conocido simplemente como wma, fue concebido como un rival al formato MP3 y al RealAudio. WMA Pro, es un algoritmo actual y avanzado, soporta audio de alta fidelidad y calidad. Igualmente existe un códec de compresión sin pérdida de datos, el cual comprime datos sin perderlos. También tiene otra variación llamada WMA Voice, dedicado a los datos de voz, que aplica una compresión y está diseñado para tasas de bits muy bajas.

- **Windows media audio profesional**

Es un algoritmo que fue especializado para quitar limitaciones en WMA estándar, soporta codificación de audio alta resolución (24bit, frecuencias altas de muestreo).

El formato WMA Pro es incompatible con WMA estándar, en donde Microsoft aprovechó la oportunidad para crear un algoritmo de alta calidad. Mientras tanto WMA estándar pierde ante MP3 después de varias pruebas, WMA Pro ocupa una posición alta (al lado de otros formatos de alta calidad) al pasar por las mismas pruebas.

- **Windows media audio sin pérdida**

Es un algoritmo matemáticamente sin pérdida de información, del mismo proveedor de Windows Media. Por ser el precursor de algoritmo sin pérdida es aceptado por la mayoría de reproductores, fue aceptado después de su lanzamiento. Compresión muy robusta de WMA se puede comparar con la WAV, una calidad muy alta de Audio entre la rápida y normal compresión.

- **Características Técnicas sobre el formato WMA**

La vida del formato WMA hace referencia tanto al archivo de audio como a sus códec, WMA que rivaliza con MP3 y RealAudio, WMA Pro, ofrece audio multicanal de alta fidelidad, WMA sin pérdida de datos, que, a diferencia de otros formatos, comprime el audio sin renunciar su calidad, y WMA de voz, que comprime contenido de voz utilizando una baja tasa de bits. Microsoft señala que la capacidad de 64 kbit/s ofrece una calidad cercana a la propia de CD, pero los audios no están de acuerdo con esta afirmación. Los archivos WMA están inmersos habitualmente en el formato (Advanced Systems Format), que incluye audio y vídeo digitales. El formato WMA soporta una velocidad de

transferencia de audio de 48 kHz con un límite de canales estéreo. En el tipo 9.1 de WMA. Microsoft incorporo audio de bajo retardo, reduciendo así la latencia en la codificación y decodificación. (Rodriguez Navarro, 2012)

1.4.2.5. ADPCM

ADPCM es un algoritmo de compresión de voz definido por la Internacional Unión de Telecomunicaciones (UIT). Se puede usar en muchas aplicaciones como digital códec para telefonía, el algoritmo ADPCM aprovecha el alto correlación entre muestras de habla consecutiva, que permite predecir valores de muestra futuros. En lugar de codificar la muestra de voz, el algoritmo codifica la diferencia entre una muestra pronosticada y la muestra de habla. Este método proporciona más compresión eficiente con una reducción en el número de bits por muestra. (Aaron, 2010)

La técnica de compresión que utiliza ADPCM es la de codificación predictiva, ampliamente utilizado para la compresión de voz. El algoritmo ADPCM emplea una predicción lineal con pérdidas que en vez de codificar las muestras de audio independientemente del PCM, utiliza la muestra anterior (o varias muestras anteriores) para predecir las muestras futuras, después calcula la diferencia entre las muestras y lo cuantifica, luego procede a la codificación de las diferencias, y sus valores previstos ,donde se representa el nivel de cuantificación para obtener la muestra de audio reconstruida, multiplicando el número niveles de cuantificación del tamaño de paso usado por el cuantificador.

Este método es eficaz porque el tamaño de paso se actualiza constantemente, tanto el cuantificador como el des cuantificador en respuesta a las variaciones en la magnitud de las muestras de entrada, esta técnica es mucho más eficaz cuando se utiliza para la compresión de voz, ya que esta técnica de compresión es un algoritmo de impresión con pérdidas, pero estas pérdidas en la compresión de voz no son significativas. En la compresión de audio de alta fidelidad existen técnicas más eficientes, a continuación, se observa en la figura 5, diagrama de bloque del algoritmo codificador ADPCM.

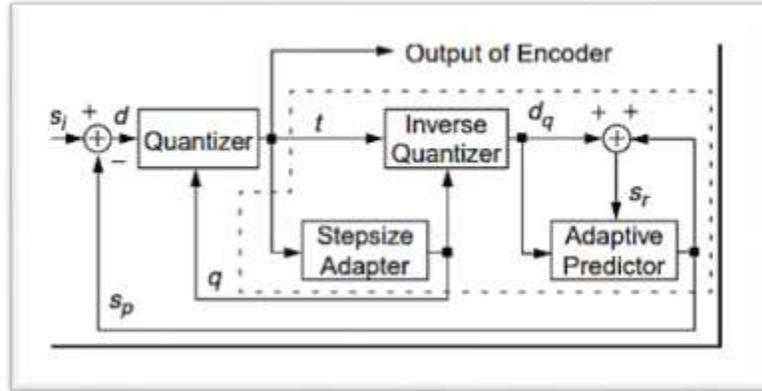


Figura 5. Codificador ADPCM

Fuente: <http://ww1.microchip.com/downloads/en/AppNotes/00643b.pdf>

Las funciones matemáticas para codificador son:

$S_i(nT_s)$ = Es la entrada de muestra.

$S_p(nT_s)$ = Es la muestra predicha.

$d(nT_s)$ = Es la diferencia entre la entrada muestreada y la salida pronosticada, a menudo llamada error de predicción.

$t(nT_s)$ = salida del cuantificador.

$S_r(nT_s)$ = Es la entrada del predictor que en realidad es la salida de verano de la salida del predictor y la salida del cuantificador.

La salida del cuantificador se representa como.

$$T(nT_s) = Q[d(nT_s)]$$

$$d(nT_s) = q(nT_s)$$

en donde

$q(nT_s)$ = es el error de cuantificación.

La entrada del predictor es la suma de la salida del cuantificador y la salida del predictor.

$$S_r(nT_s) = S_p(nT_s) + t(nT_s)$$

$$S_r(nT_s) = S_p(nT_s) + d(nT_s) + q(nT_s)$$

$$S_r(nT_s) = S_i(nT_s) + q(nT_s)$$

El mismo circuito de predicción se usa en el decodificador para reconstruir la entrada original.

Para una mejor compresión del codificador revisar anexo 1, que será enfocado en el modo de funcionamiento de la parte de codificación de del algoritmo ADPCM.

El sistema de cuantificación ADPCM está compuesto por un cuantificador adaptativo y otro predictor. El cuantificador adaptativo mide la tasa de los diferentes signos más conocidos como controles de velocidad. Este permite también un factor de escala que puede ser adaptado en los que cabe la diferencia de los niveles de codificación y actualización de los coeficientes del predictor que son usados por el algoritmo por ende El ADPCM disminuye la velocidad de bits de la voz. A continuación, se muestra en la figura 6, el diagrama de bloque de decodificador en donde reconstruye el valor original de la muestra cuantificada multiplicando el número de niveles de cuantificación por el paso de cuantificación utilizado por el cuantificador.

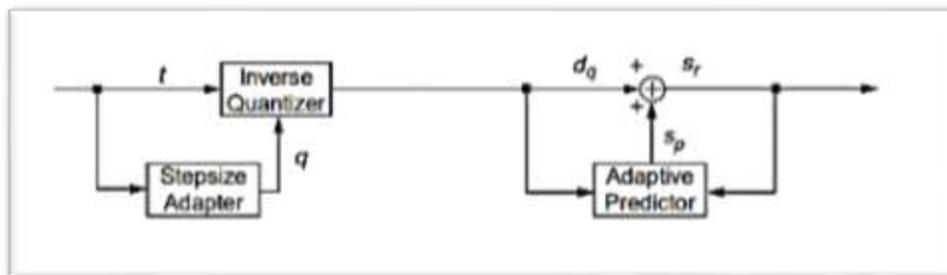


Figura 6. Decodificador ADPCM

Fuente: <http://ww1.microchip.com/downloads/en/AppNotes/00643b.pdf>

El diagrama muestra cómo se procesa la descodificación del algoritmo, se ha introducido una muestra, después de que la muestra ingresa, se calcula la diferencia del valor de entrada que se añade al anterior estimando la salida de la muestra. La suma de estos dos valores es

el valor estimado de datos que sale descomprimido, para una mejor comprensión del decodificador ADPCM revisar anexo 2.

1.5. DISPOSITIVOS INALÁMBRICOS

El dispositivo inalámbrico cumple muchas funciones con el hecho de que utilizan el medio inalámbrico como medio de transferir datos como el control remoto que por medio de emisión de ondas envía sus datos a distancias que no es necesario el cable los más usados son las impresoras, controles para dispositivos multimedia,

Existen varios tipos de conexiones inalámbricas de largo y corto alcance entre ellas hay una que se caracteriza por utilizar medios de transmisión de radio frecuencia, un alcance mayor a 10 metros, y otros dispositivos de largo alcance que se encuentran a unas frecuencias de 2.4 Ghz, y cuentan con características relevantes para dispositivos que pueden ser utilizados en lugares de difícil acceso.

Los módulos inalámbricos se los utiliza para formar una red que permita la monitorización y control de procesos de fábricas, oficinas, casas que se pueden manipulas a largas distancias. Con la implementación de dispositivos inalámbricos puede reducir el costo de la supervisión y ejecución de una fábrica, eliminando la necesidad de cables de extensión, conductos y otros accesorios costosos como se observa en la figura 7.



Figura 7. Sensores Inalámbricos

Fuente: <https://migueysanchez.wordpress.com/2015/09/29/redes-de-sensores-inalambricos-sector-cuaternario/>

1.5.1. ZIGBEE

Zigbee es una tecnología inalámbrica desarrollada como un estándar global abierto para abordar las necesidades únicas de las redes inalámbricas, de bajo costo y reducido gasto de energía. El estándar Zigbee funciona en la determinación de radio física IEEE 802.15.4 y opera en bandas libres, incluyendo 2,4 GHz, 900 MHz y 868 MHz. El protocolo Zigbee 3.0 está diseñado para comunicar datos a través de entornos de RF ruidosos que son comunes en aplicaciones comerciales e industriales.

Las especificaciones de Zigbee, mantenidas y actualizadas por Zigbee Alliance, impulsan el estándar IEEE 802.15.4 añadiendo capas de red y seguridad además de un marco de aplicación.

El estándar ZigBee fue diseñado con las siguientes especificaciones:

- Diversas bandas de trabajo: 2,4 GHz (16 canales), 915 MHz (10 canales) y 868 MHz (1 canal)
- Direccionamiento a nivel red de 16 bits
- Soporte para múltiples topologías de red tales como redes punto a punto, punto a multipunto y malla
- Ciclo de trabajo bajo: proporciona una batería de larga duración
- Baja latencia
- Espectro de propagación de secuencia directa (DSSS)
- Hasta 65,000 nodos por red
- Encriptación AES de 128 bits para conexiones de datos seguras
- Evitar colisiones, reintentos y reconocimientos
- Soporta redes slotted (QoS) y non- slotted
- Bajo consumo energético
- Gran densidad de nodos por red
- Radio de cobertura hasta 500 m según el entorno
- Ultra bajo consumo que permita usar equipos a batería
- Bajo costo de dispositivos y de instalación y mantenimiento de ellos.
- Optimizado para ciclo efectivo de transmisión menor a 0.1 %

- Velocidad de transmisión menor que 250 kbps. Típica: menor que 20 kbp

El avance tecnológico de comunicación inalámbrica Zigbee emplea la banda ISM y por lo absoluto, se encuentra la banda 2.4GHz para interactuar con los distintos dispositivos que están formando la red.

Los diversos estándares que se pueden aplicar en redes de corto alcance como es el caso del 802.11 y Bluetooth. Este está desarrollado para una clase de aplicación determinada. ZigBee es un modelo adecuado hoy en día para usar en redes con dispositivos inalámbricos y actuadores que funciona a batería. (Gutiérrez, 2015)

1.5.1.1. PROTOCOLO ZIGBEE

Zigbee es uno de los protocolos de comunicación utilizados por las redes de sensores inalámbricos. Se basa en el estándar IEEE 802.15.4 para redes de área personal (Personal Area Networks). Pero Zigbee no es el único protocolo disponible y en muchos casos tampoco el mejor pues existe una inmensa variedad de protocolos de comunicación inalámbricas para áreas locales y de mayor extensión (Barneda Faudot, 2010).

Los dispositivos Zigbee sirven para aplicaciones que requieren baja potencia, larga duración de batería y velocidad de transferencia relativamente baja. En una red zigbee los dispositivos sensores permanecen la mayor parte del tiempo en estado de hibernación despertando periódicamente o mediante un impulso externo. Los dispositivos de una red zigbee son los nodos, repetidores y un Gateway.

Las redes zigbee se utilizan en aplicaciones de medición y control. El bajo consumo de energía permite larga duración de las baterías sin necesidad de mantenimiento. El protocolo opera en las bandas de radio no licenciadas incluyendo 868 MHz en Europa, 915 MHz en Estados Unidos y 2.4 GHz en casi todo el mundo. La velocidad de transmisión típica es de 250 Kbyte / segundo, lo que es relativamente bajo y no sirve para transmisión de video o audio.

Además del protocolo de comunicación inalámbrica propiamente tal, el estándar zigbee define ciertos perfiles de aplicación que facilitan el desarrollo de aplicaciones en las siguientes áreas:

- Domótica (Home Automation)
- Energía Inteligente 1.0 (Smart Energy 1.0)
- Telecomunicación
- Asistencia Médica
- Control Remoto
- Energía Inteligente 2.0 (Smart Energy 2.0)
- Automatización de Edificios (Building Automation)
- Servicios Retail (Retail Services)

1.5.1.2. ESTRUCTURA DEL ESTÁNDAR ZIGBEE

Se puede observar en la figura 8, los campos de los diferentes tipos de tramas: datos, ACK, MAC y baliza. Un límite de 104 bytes de información soporta la trama de datos. Esta se halla numerada para cerciorar que todas las tramas llegan, además para incrementar la fiabilidad de transmisión, debe existir una confirmación de que la trama ha alcanzado su destino sin errores.

La trama ACK es muy importante, llamada también de reconocimiento. Es enviada a partir el receptor con destino al emisor, y así permitir la confirmación que el paquete ha llegado a su destino sin faltas.

Para la configuración de la red a distancia o remotamente se usa la trama MAC, así como el control y arreglo de dispositivos y nodo.

El uso de balizas es muy importante ya que en esta tecnología los dispositivos funcionan la mayor parte del tiempo en estado pasivo, se hace necesario tramas que activen los nodos, literalmente despertando los dispositivos, los cuales si no recogen más información vuelven a dormir o a entrar en modo pasivo, de esta manera se logra ahorrar una gran cantidad de batería pasando todo el tiempo encendidos.

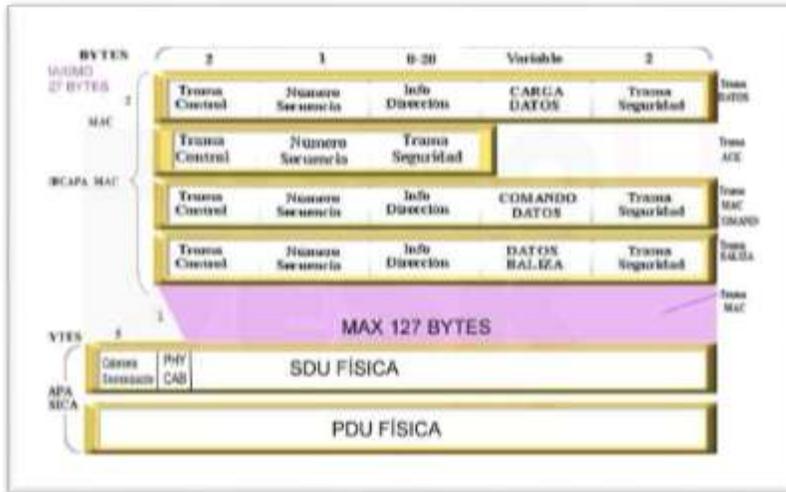


Figura 8. Formato de trama zigbee

Fuente: <https://sx-de-tx.wikispaces.com/ZIGBEE?responseToken=0a412c7fe19562dc0ecc747d6d1a38e0>

El dispositivo ZigBee característico contiene un fragmento que consta de circuito integrado de radio frecuencia RF IC acoplado a un mínimo empleo de voltaje del microcontrolador, además vinculado a un sistema de sensor o actuador. Los protocolos y aplicaciones son desarrollados sobre un chip de memoria flash.

1.5.1.3. ARQUITECTURA

La arquitectura del estándar ZigBee posee una pila de protocolos, muy parecida al modelo OSI el cual está constituido por diferentes capas, estas son independientes entre sí. Es posible observar las diferentes capas que forman parte de la pila de protocolos como se observa en la figura 9.



Figura 9. Capas del protocolo Zigbee

Fuente: <https://sx-de-tx.wikispaces.com/ZIGBEE>

La transmisión de datos por el aire, punto a punto esta proporcionada por la capa física, en conjunto con la capa de acceso al medio, además el estándar IEEE 802.15.4 hace descripción de par de capas.

- El estándar opera por medio las bandas ISM de uso no regulado, en el lugar que se precisan un máximo de 16 canales dentro del alcance de 2.4 GHz, estos canales poseen un ancho de banda de 5 MHz. En donde se puede lograr una velocidad de transmisión en el aire a valores de 250 Kbps en medidas que llegan a los 10 y 75 m, los mismos que estrictamente dependen del entorno.
- La meta principal de la capa de red (NWK) es lograr el uso adecuado del subnivel MAC y además brindar una interfaz optima que la capa de aplicación usara. NWK proporciona las técnicas requeridas como: iniciación de la red, unión a la red, enrutamiento de paquetes mandados a diferentes nodos en la red, proporcionando facilidades que permitan la transmisión del paquete al destino, filtrando paquetes recibidos, cifrar y autentificar. Hay que conocer que el códec de enrutamiento usado es la configuración de malla, la cual trabaja con el protocolo Ad Hoc On-Demand Vector Routing – AODV. Al realizar el trabajo de unir o distanciar dispositivos por medio del controlador de red, ejecuta seguridad, y orienta tramas a cada uno de sus respectivos destinos; igualmente, la capa de red del controlador de red tiene la responsabilidad de establecer una nueva red y proporcionar direcciones que posee esta. Es aquí donde se generan varias configuraciones de red que ZigBee tolera (arbol, estrella y malla).
- La capa de soporte de aplicación sostiene el rol que el nodo juega en la red, filtrando paquetes a altura de aplicación, llevando el papel que juegan en grupos y elementos con los que la aplicación trabaja y facilitando la transmisión de datos a los varios dispositivos de la red.
- La capa de aplicación que no es más que la aplicación misma y de la cual los fabricantes son los que la desarrollan. Aquí se pueden hallar los ZDO (Objetivo de dispositivo ZigBee) los cuales están encargados de definir la función del nodo internamente en la red, su configuración final; además la subcapa APS y los objetivos del sistema puntualizados por cada uno de los fabricantes.

- La comunicación entre capas es dada por medio de una aplicación de datos y otra de control, las capas que se encuentran en un nivel más alto solicitan servicios a las capas de posición inferior, y estas reportan sus resultados a las superiores. Existe además de lo mencionado otro par de módulos: módulo de seguridad, es el cual proporciona los requerimientos para cifrar y autenticar los datos, y el módulo de administración del equipo ZigBee, es el cual dirige los servicios de red del equipo local, también se encarga de proporcionar a la aplicación funciones de dirección remota de red.

1.5.1.4. TOPOLOGÍA ZIGBEE

Los datos bidireccionales de Zigbee se transfieren en dos modos: modo sin baliza y modo de baliza. En un modo de baliza, los coordinadores y enrutadores monitorean continuamente el estado activo de los datos entrantes, por lo tanto, se consume más energía. En este modo, los enrutadores y coordinadores no duermen porque en cualquier momento cualquier nodo puede despertarse y comunicarse. Sin embargo, requiere más suministro de energía y su consumo de energía general es bajo debido a que la mayoría de los dispositivos se encuentran en un estado inactivo por extensos períodos en la red.

En topologías de malla y árbol, la red Zigbee se amplía con varios enrutadores donde el coordinador es responsable de mirarlos. Estas estructuras permiten que cualquier dispositivo se comunique con cualquier otro nodo adyacente para proporcionar redundancia a los datos. Si algún nodo falla, la información se enruta automáticamente a otro dispositivo mediante estas topologías. Como la redundancia es el factor principal en las industrias, por lo tanto, la topología de malla se utiliza principalmente. En una red de árbol de clúster, cada clúster consta de un coordinador con nodos hoja, y estos coordinadores están conectados al coordinador principal que inicia toda la red (Moreno & Glen, 2012), como se observa en la figura 10.

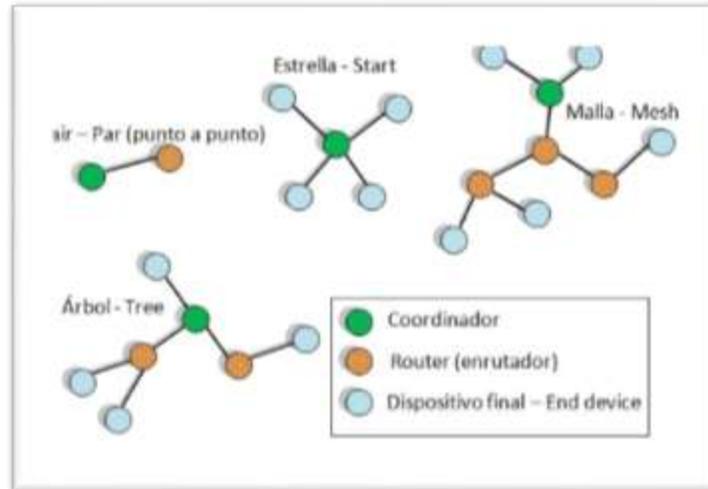


Figura 10. Topología Zigbee

Fuente: <https://sx-de-tx.wikispaces.com/ZIGBEE?responseToken=0a412c7fe19562dc0ecc747d6d1a38e0>

- **TOPOLOGÍA DE ESTRELLA**

En una configuración de red estelar o estrella, hay un coordinador y cualquier cantidad de dispositivos finales. Todos los dispositivos finales están conectados al coordinador y los dispositivos finales individuales están aislados, tanto física como eléctricamente, es decir, no hay conexión directa entre los dispositivos finales.

- **TOPOLOGÍA DE MALLA**

La topología Zigbee Mesh también es una topología punto a punto y es una extensión de la topología del árbol de clúster. Los dispositivos finales que están configurados como FFD pueden comunicarse directamente con otros dispositivos FFD (enrutadores o dispositivos finales). Pero los dispositivos finales configurados como RFD aún necesitan comunicarse a través de enrutadores o coordinadores.

- **TOPOLOGÍA DE ÁRBOL**

La topología de árbol de clúster es un modelo de topología de igual a igual. En Zigbee Cluster Tree Topology, los dispositivos finales se unen a la red a través del coordinador o el enrutador. Como el enrutador Zigbee amplía el alcance de la red Zigbee, el dispositivo final no necesita estar dentro del alcance del coordinador. (Adarve, 2015)

Cada uno de estos dispositivos ha sido diseñado para propósitos distintos. El dispositivo de función reducida desarrollado para aplicaciones muy básicas, como interruptores de iluminación y sensores infrarrojos, donde no se requiere enviar o recibir extensas cantidades de información. Únicamente es capaz de comunicarse con dispositivos FFD. Esto hace posible que se logre implementar aprovechando los mínimos recursos posibles, así como un ahorro energético visible. En cambio, los FFDs pueden actuar como coordinadores o como dispositivos finales. Pueden comunicarse con otros FFDs y RFDs. Para ello necesitan más recursos, han de implementar la pila completa y precisan un consumo más exigente.

1.5.1.5. SEGURIDAD

Existen varias limitaciones para aplicar una seguridad más eficiente dentro de ZigBee, aun así, se definen medidas básicas de seguridad efectuados en la subcapa MAC para garantizar el buen funcionamiento y la interoperabilidad de dispositivos. Pero aquellos parámetros de seguridad no precisamente estarán aplicados en todos los dispositivos y en todo momento. Las capas superiores serán las que establezcan en qué momento se implementara la seguridad en la subcapa MAC. Los servicios que se pueden ofrecer en el modo seguro son:

- Control de Acceso
- Encriptación de Datos
- Integridad de Trama
- Refrescamiento Secuencial

• CONTROL DE ACCESO

Es una prestación de seguridad que ofrece la capacidad a un nodo de poder elegir otros con los cuales pueda comunicarse. Un dispositivo mantiene una lista los dispositivos o nodos con los cuales puede esperar tramas de datos.

• ENCRIPCIÓN DE DATOS

En Zigbee la encriptación de datos es una prestación de seguridad que usa una cifra simétrica para salvaguardar los datos de ser leídos por otros dispositivos. La información puede ser encriptado usando una clave compartida por un grupo de dispositivos o usando una clave compartida entre dos pares. La encriptación puede ser de la trama de datos.

- **INTEGRIDAD DE TRAMAS**

Es una prestación de seguridad maneja Códigos de Integridad de Mensajes (MIC), para resguardar datos que no poseen clave criptográfica. La integridad puede ser dada a tramas de datos, beacons y comandos. La clave usada para dar integridad de trama puede ser compartida por un grupo de dispositivos o entre dos pares.

- **MODO SEGURO**

Al operar en modo seguro logran suministrar de cualquiera de los servicios de seguridad detallados anteriormente. Los servicios ofrecidos dependen de la configuración de seguridad usada.

1.5.1.6. APLICACIONES DE ZIGBEE

Zigbee Networking y Zigbee Technology tiene una amplia gama de aplicaciones como Domótica, Atención médica y Seguimiento de materiales. Veamos algunas aplicaciones de Zigbee Technology, donde los dispositivos Zigbee pueden aumentar la eficiencia y reducir los costos. Se lo utiliza para control industrial, juntar sensores incrustados, toma datos en hospitales, realiza labores de impedimento de humo o intrusos. La formación de la red utilizará una mínima cantidad de energía con esto cada dispositivo inico pueda tener una libertad de hasta 5 años antes de necesitar un suplemento en su sistema de alimentación, como se puede observar en la siguiente figura 11.

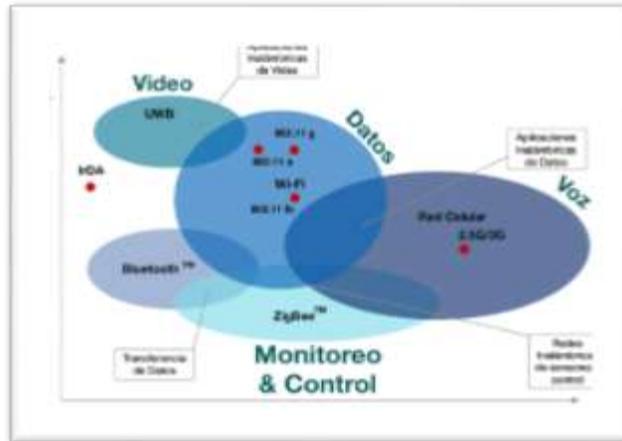


Figura 11.Aplicaciones de red Zigbee

Fuente:<https://sx-de-tx.wikispaces.com/ZIGBEE?responseToken=0a412c7fe19562dc0ecc747d6d1a38e0>

1.5.2. XBEE

Es el nombre comercial de una familia de módulos de comunicación por radio y están basados en el estándar zigbee. Existen muchos módulos Xbee basados en el estándar IEEE 802.15.4 Los módulos Xbee han sido esquematizados para aplicaciones que requieren de un tráfico de datos muy extenso, con una latencia muybaja y una concordancia de comunicación predecible. Zigbee proporciona que otros dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Es especialmente empleado para redes de sensores en entornos industriales, médicos y, sobre todo, domóticas. (Durán Acevedo & García Sierra, 2013)

1.5.2.1. XBEE S1

Un módulo integrado XBee es realmente tres cosas en uno, primero un factor de forma, una interfaz de host y un grupo de diferentes protocolos. Está fabricado para una variedad de usos, desde ZigBee hasta aplicaciones de baja latencia de alto rendimiento. Una cosa buena de Xbee es que puede hacer cualquier cosa que ZigBee pueda hacer (si usa la versión de ZigBee), pero también incluye otras pilas de protocolos en el mismo factor de forma y la misma interfaz de host. Por ejemplo, la radio XBee sub-GHz le permite tener un alcance mucho más largo que una radio ZigBee de 2.4 GHz., para mayor información técnica del dispositivo. El factor de forma se refiere a una configuración de pin y forma común en el

hardware. Tiene un zócalo de 20 pines o 37 pines (dependiendo de la huella de hardware que seleccione), que ocupa una cantidad muy pequeña de espacio en la placa de circuito. Esta configuración hace que sea más fácil para usted traer un nuevo dispositivo al mercado, porque si el factor de forma es el mismo, la interfaz del host es la misma se adjunta en el anexo 3.



Figura 12. Xbee S1

Fuente: <https://aprendiendoarduino.wordpress.com/2016/11/16/zigbeexbee/>

1.6. RED (Wireless Sensor Networks)

Una red de sensores inalámbricos (WSN) es la que está formada por una gran cantidad de nodos donde cada nodo está equipado con un sensor para enviar datos de distintos dispositivos que cuentan con sensores como la luz, el calor, la presión. Los dispositivos inalámbricos vuelven menos aplicables el monitoreo a pacientes y posible el cuidado de la salud. Una red de sensores se compone de una gran cantidad de nodos de sensores, que están desplegados densamente dentro del fenómeno o muy cerca de él. La posición de los nodos sensores no necesita ser diseñada o predeterminada. Esto permite un despliegue aleatorio en terrenos inaccesibles o en operaciones de socorro. Por otro lado, esto también significa que los protocolos y algoritmos de redes de sensores deben poseer capacidades de auto organización. Otra característica única de las redes de sensores es el esfuerzo cooperativo de los nodos sensores. Los nodos sensores están equipados con un procesador incorporado. En lugar de enviar los datos brutos a los nodos responsables de la fusión, los nodos sensores utilizan sus capacidades de procesamiento para realizar localmente cálculos

simples y transmitir solo los datos requeridos y parcialmente procesados (Niels & Jan, 2006).

La comunicación inalámbrica en usos industriales tiene muchas ventajas. Asimismo, de una mayor fiabilidad, la ventaja más reconocida es el menudo coste de instalación como podemos observar en la figura 13, realizando una red los dispositivos inalámbricos.

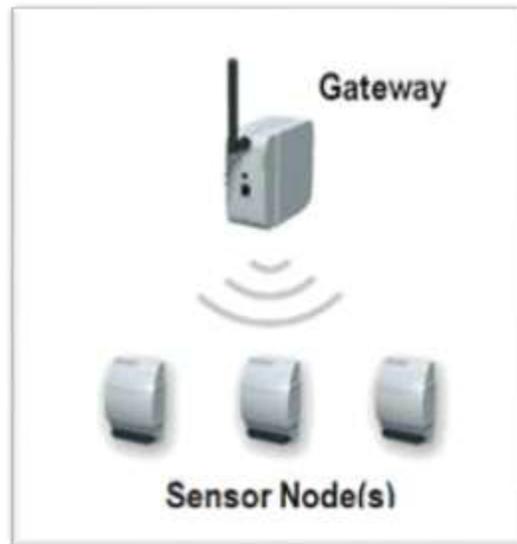


Figura 13. Red de sensores wsn
Fuente: <http://www.ni.com/white-paper/7142/es/>

1.6.1. TOPOLOGÍA

Los nodos que forman una red WSN están estratégicamente organizados en uno de los tres tipos de topologías de red de distintos forma.

- Topología de estrella, cada uno de los nodos se enlaza directamente al gateway.
- Topología de árbol, cada uno de los nodos se enlaza a un nodo de mayor jerarquía y después se conecta al gateway, los datos recolectados son ruteados desde el nodo de menor jerarquía en el árbol hasta llegar al gateway.
- Las redes de tipo malla, este tipo de red se caracteriza que los nodos se pueden conectar a múltiples nodos en el sistema y enviar los datos por el camino de mayor confiabilidad.

1.7. DISPOSITIVO PROGRAMABLE ARDUINO

El dispositivo Arduino, que para su manipulación es muy fácil tal como muestra la Figura 14, está basado en tecnología ATmega328, este dispositivo va a funcionar conjuntamente con la red de los dispositivos inalámbrico xbee para poder convertir la señal analógica a digital y enviar los datos.

Las características del dispositivo es que consta de 14 pines digitales que pueden ser usados como entradas y salidas, usando las funciones pin Mode, digital Write, y digital Read. Operan a 5 voltios. Cada pin puede proveer o recibir un máximo de 40mA y poseen una resistencia de pull-up (desconectada por defecto) de 20 a 50 kOhms. Además, algunos pines poseen funciones especializadas que nos ayudan a facilitar la manipulación del dispositivo programable.

Con el dispositivo programable arduino se de utilizar el software adecuado y descargar las librerías de acuerdo al dispositivo que vaya a utilizar para mayor comodidad en la programación.



Figura 14. Arduino Nano

Fuente: <https://store.arduino.cc/usa/arduino-nano>

1.8. MÓDULO DS3231

El módulo de tarjeta SD es especialmente útil para proyectos que requieren registro de datos como se puede observar en la figura 15, en donde se puede crear un archivo en una

tarjeta SD para escribir y guardar datos usando la biblioteca del dispositivo utilizando el protocolo de comunicación SPI.

El Módulo de Tarjeta SD tiene seis pines, dos para alimentar el módulo, los pines VCC y GND, y cuatro pines más para la comunicación SPI. Así es como se puede conectarlo a la Junta Arduino. Las tarjetas de almacenamiento se han convertido en un estándar, desplazando a otros medios de que cumplen las mismas funciones por su gran capacidad y pequeño tamaño. Por este motivo han sido añadidos a una gran cantidad de dispositivos, siendo en la actualidad componentes frecuentes en ordenadores, tabletas y Smartphone, entre otros.

Con los módulos SD se almacena los datos que es enviad desde los dispositivos programables arduino, que a su vez los datos son extraídos para luego ser procesada o aplicada el algoritmo de compresión más adecuado al proyecto.



Figura 15. *Modulo sd-card*

Fuente: <https://www.luisllamas.es/tarjeta-micro-sd-arduino/>

Su uso es evidente y frecuente con dispositivos programable, pues les ayuda en guardar o almacenar datos que tiene que procesar o realizar algún análisis, y los módulos SD-card es la mejor opción es estos casos.

CAPITULO II

2. METODOLOGÍA

2.1. TIPO DE ESTUDIO

Investigación de trabajo descriptivo y explicativo, es descriptiva porque se basa en la afirmación de un fenómeno de transmitir datos en red de sensores inalámbricos xbee por su limitada velocidad de transición, permite enviar datos comprimidos de audio. Y que se realiza pruebas para comprobarla factibilidad y aplicar este tipo de tecnología para la transición y recepción de datos en la Universidad Nacional de Chimborazo, analizar varios factores que puedan ayudar a mejorar el rendimiento de los diversos sistemas códec de compresión, por este motivo se la ubica en una investigación descriptiva.

Investigación explicativa: se la realiza porque se va a detallar y explicar todos los pasos y procesos del sistema a efectuar para lograr el propósito de este documento.

2.2. MÉTODOS, TÉCNICAS E INSTRUMENTACIÓN

2.2.1. Métodos

2.2.1.1. Analítico/Deductivo

Se utiliza el método analítico-deductivo en el proyecto por la razón de que se inicia con un análisis particular del funcionamiento de los componentes electrónicos necesarios para el desarrollo del sistema y la forma de interactuar entre sí para que el sistema desempeñe su función de la manera más óptima.

2.2.2. Técnicas

2.2.2.1. Observaciones

Se utilizó la técnica ya que consiste en visualizar y captar mediante los datos recolectados, en forma sistemática. Y además es estructurada porque se utiliza una guía diseñada previamente, en la que se especifican los elementos que serán observados en este proyecto, se observa los siguientes parámetros la compresión de datos, velocidad de transmisión, la potencia y el tiempo de envío y recepción de datos comprimidos.

2.2.2.2. INSTRUMENTACIÓN

Se utilizó, folletos, libros, revistas, páginas web, datasheet y otros instrumentos necesarios para la selección de la información que se refiere al diseño e implementación del proyecto.

2.3. POBLACIÓN

La población son las personas que utilizan los dispositivos como medio de comunicación, en la Universidad Nacional de Chimborazo que será evaluado la transmisión de datos a distintas distancias que por lo general necesiten poder recibir o transmitir audio con una calidad aceptable.

2.4. MUESTRA

Como la muestra tenemos las pruebas realizadas en el coliseo del prototipo de la transmisión de datos de voz.

2.5. HIPÓTESIS

El estudio y evaluación de algoritmos de compresión de audio que permite determinar un algoritmo eficiente para la recepción y transmisión de datos por una red xbee a unas diferentes distancias.

2.6. OPERACIONALIZACIÓN DE VARIABLES

Tabla 2. Operacionalización de variables

VARIABLES	CONCEPTO	CATEGORIA	INDICADORES	TECNICAS E INSTRUMENTOS.
INDEPENDIENTE Algoritmo de compresión de audio.	Es la reducción de datos originales a datos más reducidos y su ancho de banda se disminuye, para transmitir y recibir datos sin mayor dificultad.	Modelo matemático	Algoritmo de compresión con pérdida.	Análisis de los tipos de compresión.
Dispositivos inalámbricos	Son mecanismos que han brindado la facilidad de uso por la manera que trabajamos y compartimos información, su costo es bajo y bajo consumos de energía.	Tecnología	Transmisión de datos a través de sensores. Conectividad con todos los dispositivos conformados las red	Recolección de datos recibidos en una transmisión. Valoración del funcionamiento en una red
DEPENDIENTES Transmisión y recepción de datos de voz	Configurado a una misma velocidad y un mismo protocolo de comunicación entre los dispositivos.	Datos enviados	Datos entregados con la menor pérdida posible. Datos adquiridos con mayor calidad y eficiencia.	Recolección de datos de potencia de radiofrecuencia.

Fuente: Autor

2.7. PROCEDIMIENTO

Para el desarrollo y análisis del algoritmo que comprima audio con una mínima pérdida para su transmisión, se siguió una serie de procesos, que, en conjunto cumplirán con el objetivo deseado. A continuación, se presenta los siguientes pasos a seguir en la investigación propuesta:

- Búsqueda de información de algoritmos de compresión
- Análisis de los diferentes tipos de algoritmos de compresión.
- Elección del algoritmo determinado para el proyecto por sus características y su facilidad en envío por una red de dispositivos xbee.
- Programación en los módulos Atmega328.
- Aplicar el algoritmo de compresión seleccionado y estudiado en los módulos arduino.
- Configuración de protocolo de comunicación en los dispositivos inalámbricos para formar la red.
- Selección del área donde se realiza la red con los dispositivos xbee
- Transmisión de voz por medio de la red de dispositivos inalámbricos.
- Pruebas de la transmisión de voz realizadas sobre la red xbee.

2.8. PROCESAMIENTO Y ANALISIS

Para realizar todo el procedimiento del proyecto, se realizó un diagrama con todas las partes fundamentales que nos ayuda con la elaboración de parámetros como se muestra en la figura 16, se puede observar el diagrama de bloques todo el sistema.

Las partes está compuesta por un bloque llamado circuito transmisor que consta con dispositivos programables y configurables en donde la señal de la voz es transformada y comprimida para luego ser enviada al segundo bloque en donde se tiene el circuito

repetidor que envía los datos al bloque final que es el circuito receptor, en donde los datos llegan y son procesados y convertidos en audio.

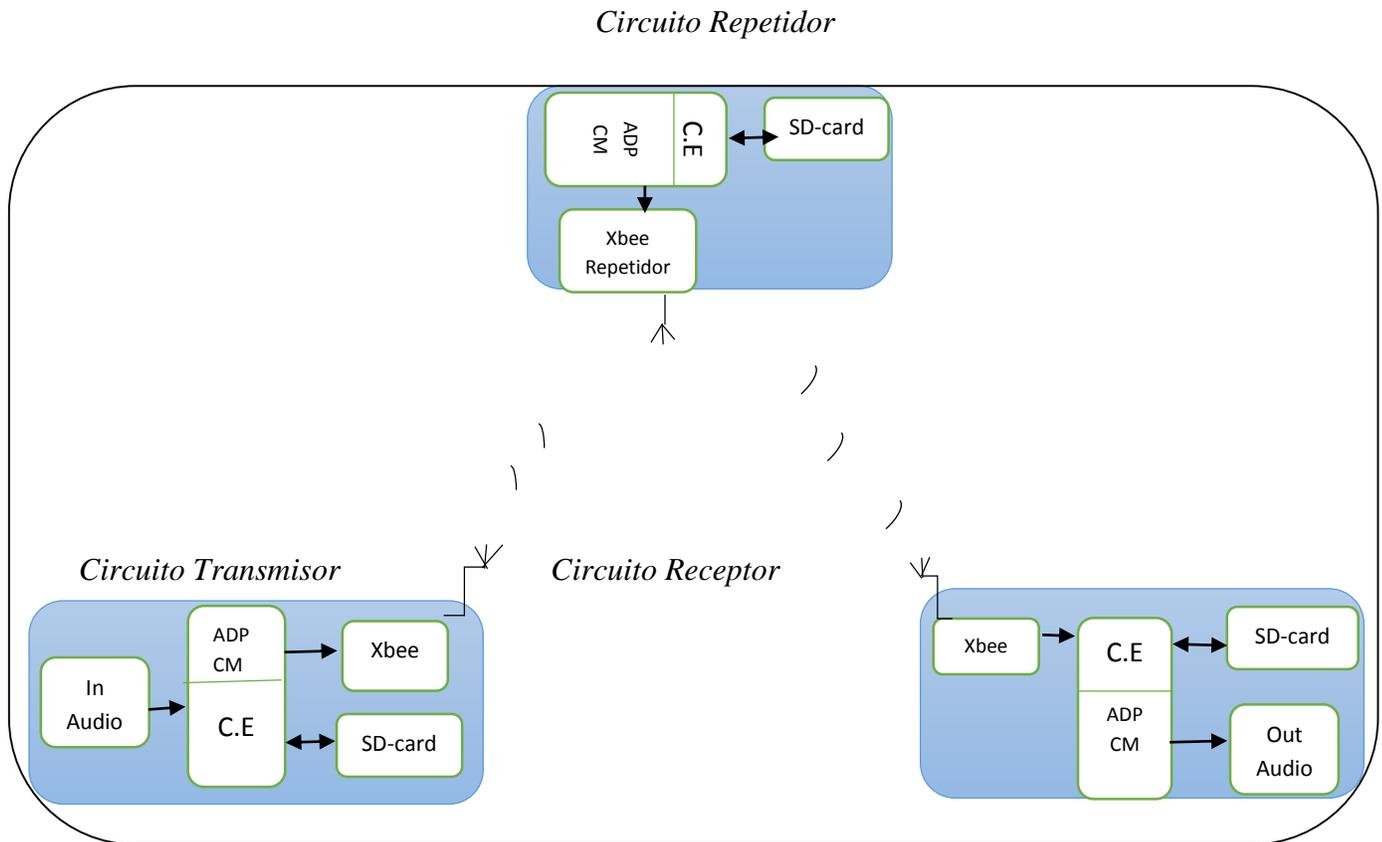


Figura 16. Diagrama de Bloque del sistema
Fuente. - Autor

2.8.1. Parámetros de evaluación de algoritmos de compresión

Se ha considerado los principales parámetros para la obtención de los resultados que muestran una eficiencia al comprimir voz y audio para su transmisión por una red de sensores inalámbricos xbee.

- Capacidad de compresión
- Calidad de compresión

2.9. ANÁLISIS DE LOS ALGORITMOS DE COMPRESIÓN DE AUDIO

Este capítulo describe las diversas pruebas realizadas en los diferentes algoritmos de compresión de voz y audio, dichas pruebas fueron realizadas en software Matlab para sus respectivos análisis y determinar el algoritmo eficiente.

El análisis se realizó en los siguientes algoritmos:

- FLAC
- AAC
- MP4
- OGA
- OGG
- ADPCM

2.9.1. CAPACIDAD DE COMPRESIÓN

En la figura 17, se observa como es el procedimiento de la compresión de datos que los algoritmos realizan.

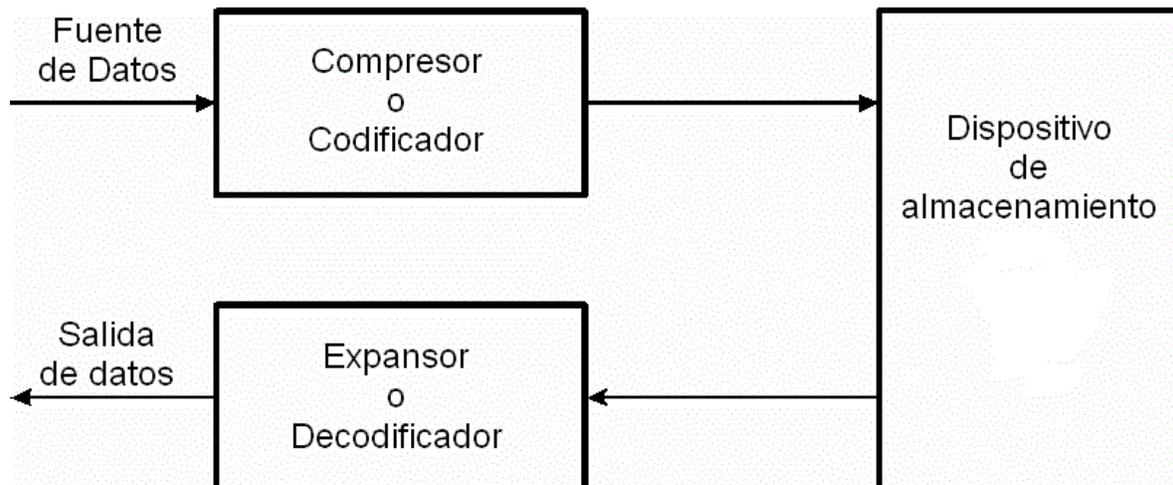


Figura 17. Compresión de datos
Fuente: Autor

Para la comparación de los algoritmos, se utilizará la voz en formato WAV ya que es uno del formato de audio sin pérdida, que es el más utilizado para almacenar sonidos y es un formato sin ningún tipo de compresión de datos.

Esta novedosa técnica consiste en convertir la voz sin comprimir en datos comprimidos con el objetivo reducir el tamaño del archivo y que luego al enviarse por cualquier medio inalámbrico, el receptor pueda recuperar una versión muy cercana al audio original. (Barzola, Cabrera, & Chávez, 2014).

Dependiendo del algoritmo y nivel de compresión seleccionado, se puede lograr una variedad de tamaños de archivo.

Como se puede observar en la figura 18, los valores obtenidos en la compresión de la voz durante 10 segundos en el software Matlab con los diferentes algoritmos de compresión.

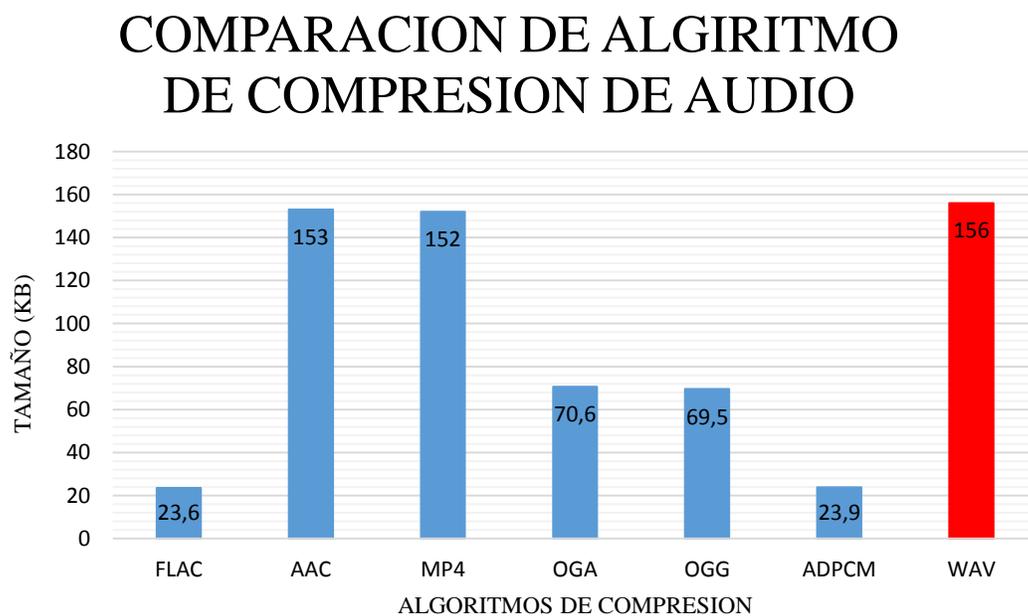


Figura 18. Compresión de varios Algoritmos
Fuente: Autor

Los algoritmos de compresión son comparados por un audio en formato wav que su tamaño está sin comprimir y se realiza la compresión con los tipos de algoritmos que se mencionó anteriormente y obtiene los distintos valores de compresión.

2.9.2. CALIDAD DE COMPRESIÓN

El formato de audio más popular es el Mp3 de los varios formatos de audio que se puede utilizar como flac, wma, ogg entre otros, cuentan con características que las diferencian de uno al otro.

Los parámetros que se tomó en cuenta para realizar el análisis de los algoritmos de compresión es la velocidad de transmisión (Bitrate) que se mide en Kbps (kilobits por segundo), siendo 128 kbps la calidad de compresión estándar que es iguala al sonido de radio Fm, en donde a mayor Bitrate la calidad de audio es más alta y su tamaño crece.

En la figura 19, tenemos varios algoritmos de compresión que fue aplicado en un audio de formato Wav algoritmo sin compresión, en donde se observa su velocidad transmisión de cada algoritmo.

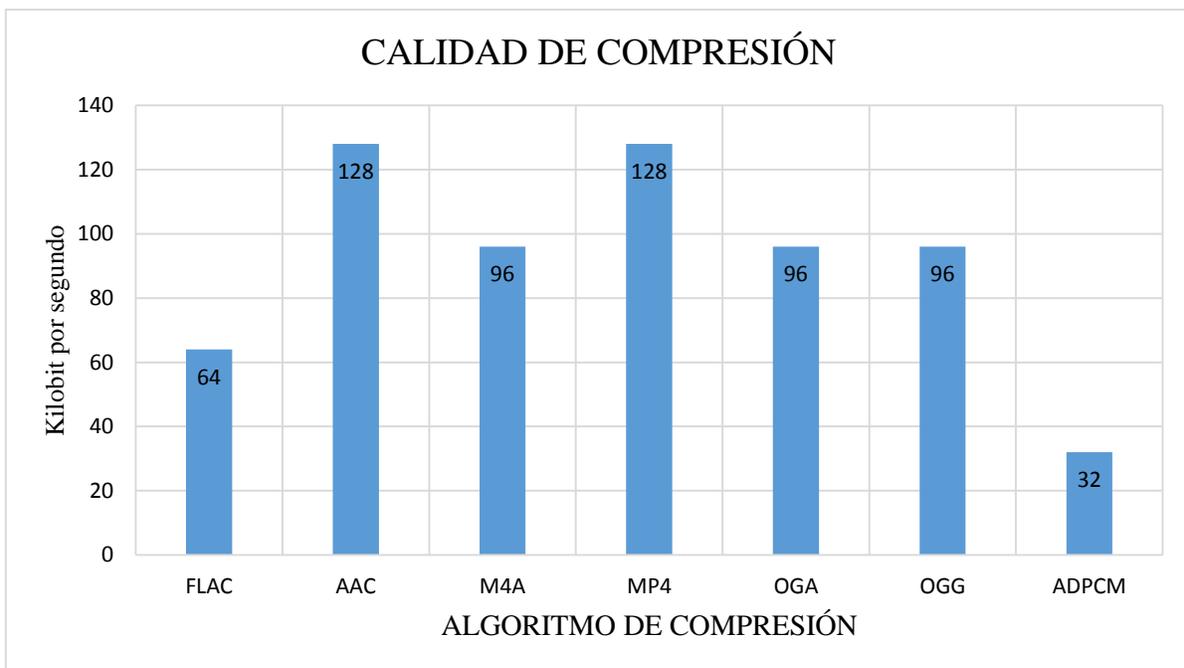


Figura 19. Calidad de compresión
Fuente: Autor

De acuerdo a los análisis realizados con instrucciones en el software Matlab en el primer parámetro se determinó que el algoritmo de compresión **Flac** es el que mejor estado de

compresión determina y en el segundo parámetro el algoritmo **Mp4** y **AAC** son los de mayor calidad y a su vez de mayor tamaño, tiene el archivo al momento de comprimir, pero por las características que se necesita para el dispositivo desarrollador al ocupar un micro controlador Atmega 328 y por sus características el más adecuado es algoritmo **ADPCM** que se adapta mejor al micro controlado por lo tanto se solucionó con el problema de algoritmo.

2.9.3. Diseño del Circuito Transmisor

El circuito transmisor está formado por cinco etapas como se muestra en la figura 20, la primera etapa está compuesta por el micrófono, que permite captar la voz e introducirla al circuito por medio de voltajes y como segunda etapa se tiene el circuito sujetador que nos permite elevar el nivel de voltaje y la tercera etapa que introduce la señal al dispositivo Arduino nano que convierte la señal analógica en señal digital que será procesada y comprimida con el algoritmo ya seleccionado, como cuarta etapa se tiene una tarjeta sd card en donde la información será almacenada en forma de datos y como quinta etapa se tiene el modula xbee que permite enviar la información.

En la figura 20 se observa el diagrama de bloques del sistema transmisor.

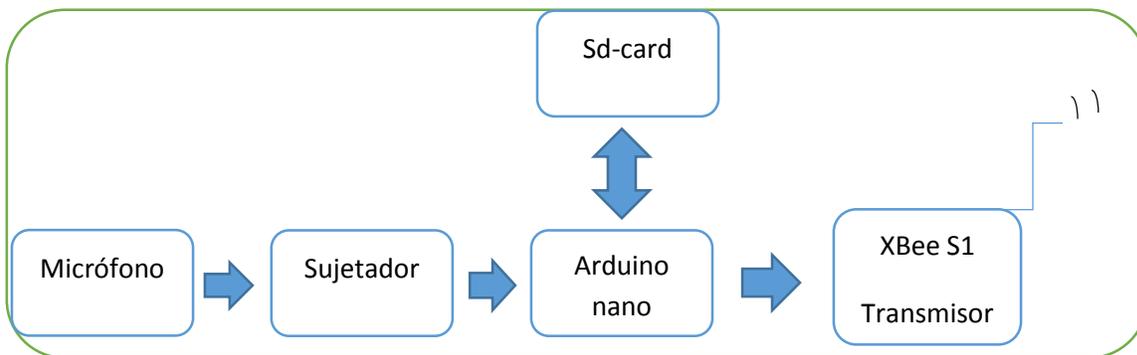


Figura 20. Diagrama de Sistema Transmisor

Fuente: Autor

➤ **Micrófono**

Como primera etapa tenemos el micrófono por donde se introduce la señal de la voz en forma de onda sonora y las transforma en energía eléctrica para ser luego procesada.

➤ **Circuito Sujetador**

En la gráfica 23 se tiene el simulado del circuito que cuenta con dos resistencias de 10 k, configurado como divisor de voltaje para obtener 2.5 voltios, que se suma a la señal de audio, para obtener una señal entre 0 y 5 voltios que accede a una entrada analógica de un dispositivo Arduino nano para ser procesada y aplicada la compresión como se observa en la figura 21.

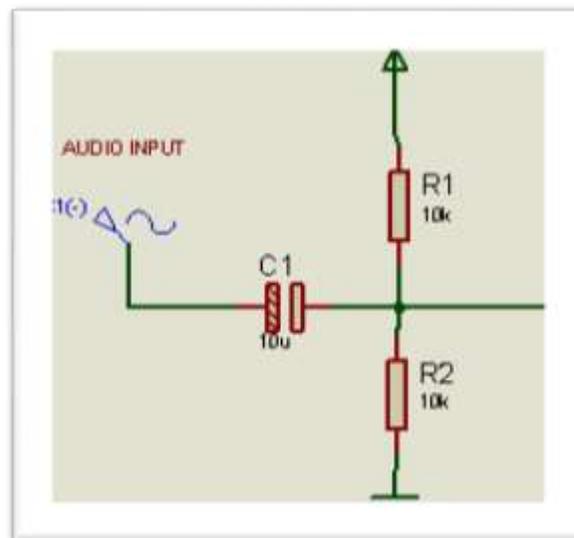


Figura 21. Circuito sujetador
Fuente: Autor

➤ **Arduino Nano**

En la figura 22 se observa que el arduino nano recibe por su pin analógica la señal de audio ajustada anteriormente para ser convertida en señal digital que será procesada por el Atm 328p, la información digitalizada es almacenada en el módulo sd-card, que se conectan por medio de los pines de comunicación con el arduino, además estará encargada de enviar la

información almacenada desde el modulo sd-card al dispositivo Atmega328p, que a su vez envía los datos comprimidos por el xbee por el pin 1 (tx).

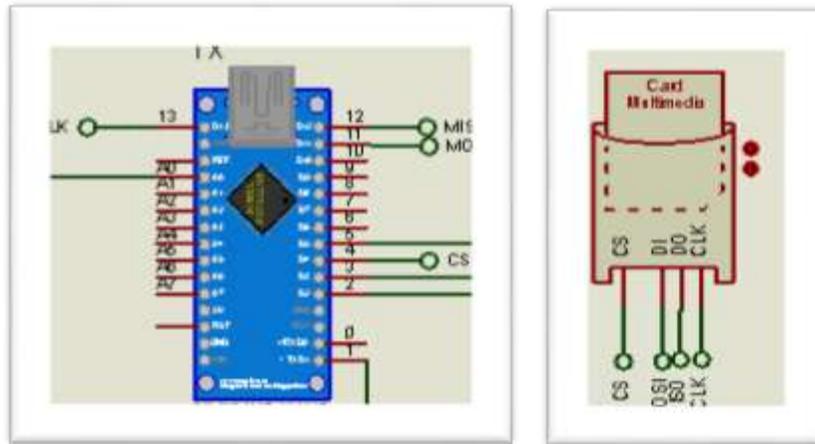


Figura 22. Arduino Nano y sd-card
Fuente: Autor

➤ Xbee S1

En la figura 23 se observa el modula Xbee Coordinador recibe los datos del arduino Atmega328 ya comprimidos y a una velocidad ya establecida en la programación, para realizar él envío de datos sobre la red, del xbee coordinador al dispositivo xbee router que trabaja como repetidor que se encuentra a una distancia máxima y con una velocidad de 56700 baudios en un mismo canal de frecuencia ya configurada anteriormente.

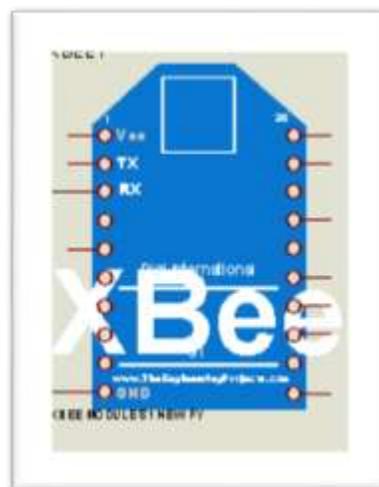


Figura 23. Dispositivos Xbee
Fuente: Autor

En lo siguiente se tiene todo el circuito electrónico y sus componentes que fue realizado en el software de simulación Proteus 8.0, en donde se conectan los elementos que son utilizados en el circuito transmisor que cada componente cumple con su objetivo y poder realizar el proyecto de una mejor forma como se observa en la figura 24, la simulación de todo el circuito transmisor.

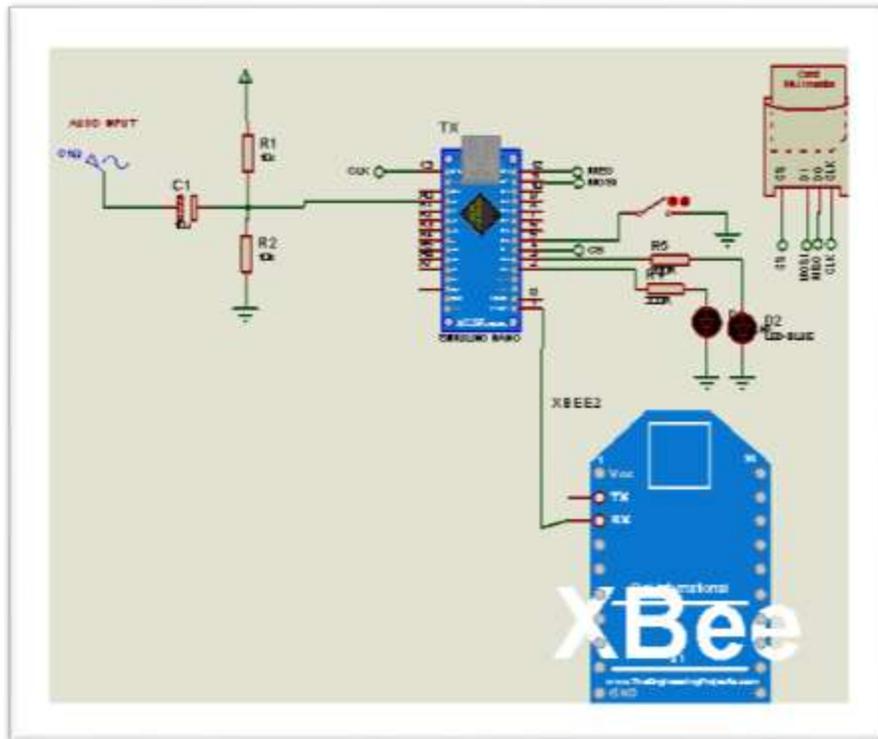


Figura 24. Circuito electrónico transmisor
Fuente: Autor

➤ Realización de placas del prototipo

Para realizar el prototipo del proyecto fue necesario la elaboración de todas las placas y por ende se utilizó el software de simulación proteus, la primera placa es la del circuito transmisor que fue realizada con las dimensiones de todos los elementos, para ya realizar la placa y visualizar en3D y con todas las conexiones como se puede apreciar en las figuras 25.

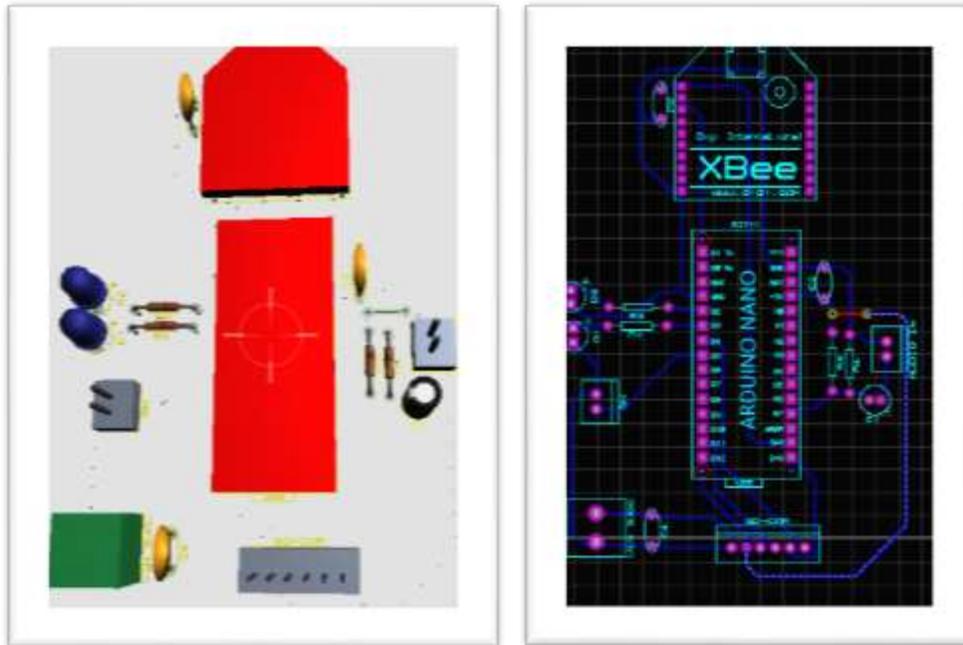


Figura 25. Circuito Transmisor en 3 y en ARES -Software Proteus
Fuente: Autor

➤ Descripción del código de Programación

La programación fue realizada en dispositivo arduino Atmega 328p, utilizando las librerías de dispositivo nano V3y que se describe en el diagrama de flujo de la figura 26, del circuito transmisor y para mayor análisis se adjunta en el anexo 4.

En donde se encuentra la programación detallada de todo los códigos y comandos que se utilizó de todo el circuito transmisor para ser enviado al siguiente dispositivo que cumple la función de recibir y reenviar los datos.

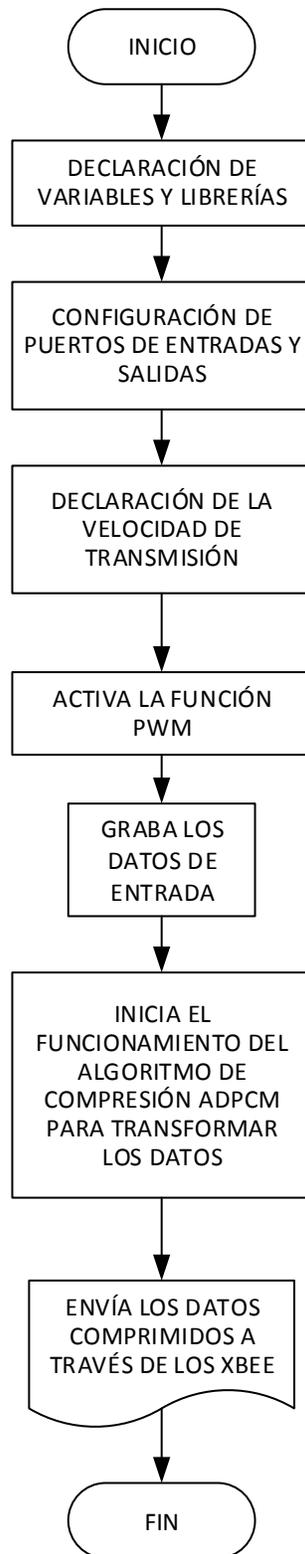


Figura 26.Diagrama de Flujo del Transmisor

Fuente: Autor

2.9.4. DISEÑO DEL CIRCUITO REPETIDOR

Para el diseño del circuito repetidor tenemos que realizar varias etapas que consta como primera fase el dispositivo inalámbrico xbee que cumple la función de recibir los datos comprimidos por el pin rx y de enviar por el pin tx, para que el siguiente xbee, pero también se le puede reproducir añadiendo el circuito eléctrico con el algoritmo y el módulo sd-card que reciba todos los datos sin perderlos en la trayectoria, como se puede observar en la siguiente figura 27.

Diagrama de bloques del sistema repetidor.

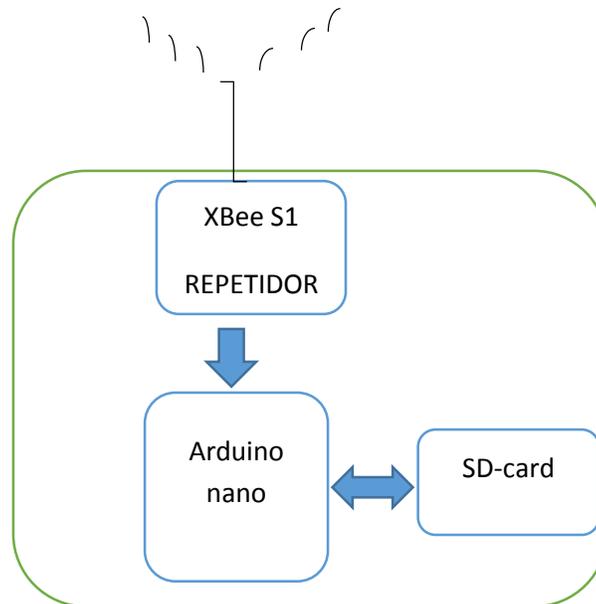


Figura 27. Diagrama Repetidor

Fuente: Autor

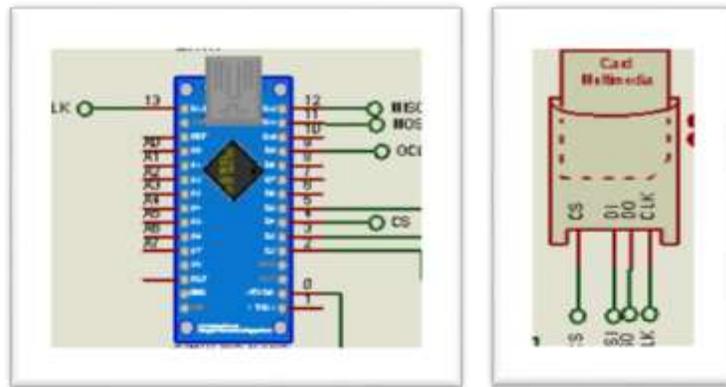
➤ **Xbee Repetidor**

El dispositivo inalámbrico recibe los datos comprimidos del dispositivo inalámbrico xbee coordinador por el pin rx y lo vuelve a reenviar todos los datos por el pin tx y por la red hacia el dispositivo destino final.

➤ **Arduino Nano**

El arduino nano recibe los datos por su pin de entrada (rx) del módulo xbee, la información es descomprimida y almacenada en un módulo sd-card, además estará encargada de

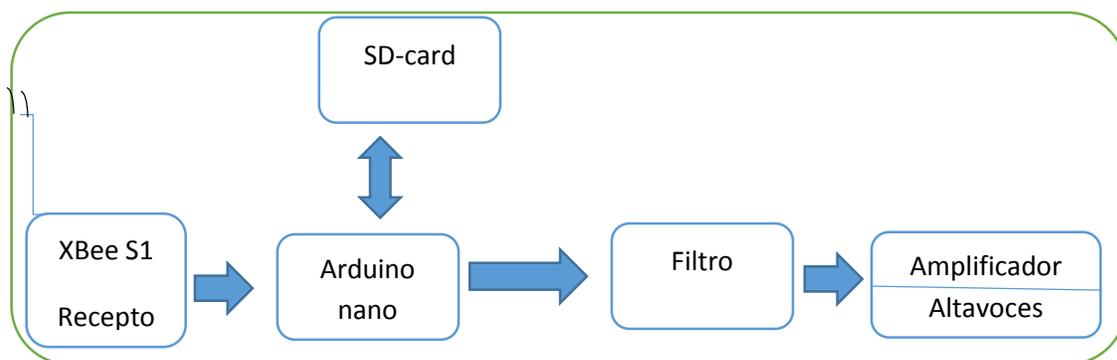
convertir la señal digital en analógica y de enviar la información almacenada para ser reproducida, como se muestra en la figura 28.



*Figura 28. Arduino nano y el modulo sd-card
Fuente: Autor*

2.9.5. DISEÑO DEL CIRCUITO RECEPTOR

Para el diseño del circuito receptor, se cuenta con cinco etapas, tenemos la primera etapa que consta de un dispositivo xbee que recibe la información de datos comprimidos, como segunda etapa tenemos el modulo sd-card que almacena la información en un micro sd y en tercera etapa se tiene el arduino nano que transforma la información de datos comprimidos en datos digital y en señal analógica y como cuarta etapa se tiene el filtro pasa bandas que elimina ruidos no deseados y como quinta etapa se tiene los parlantes con un pre amplificador que recibe la información y la reproduce el audio como se observa en la figura 29 el diagrama de bloques del sistema receptor.



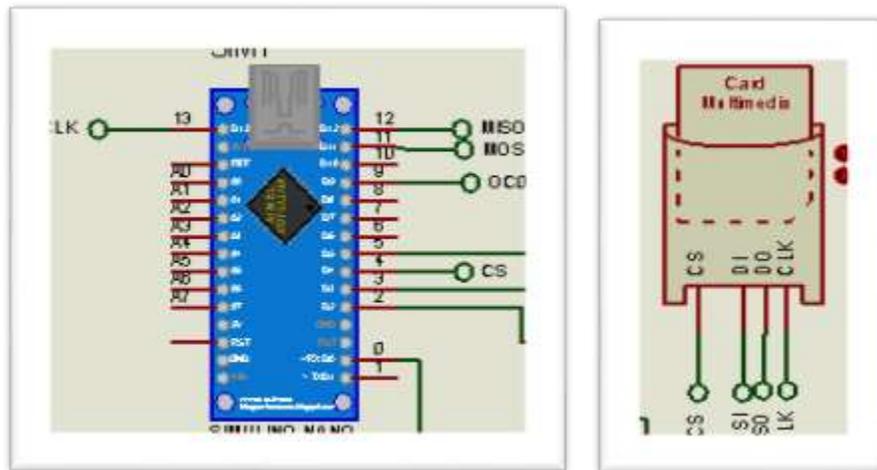
*Figura 29. Diagrama del Sistema Receptor
Fuente: Autor*

➤ **Xbee s1**

El módulo Xbee destino final recibe los datos del módulo Xbee router que funciona como repetidor, que recibe los datos comprimidos por el pin rx con una velocidad de transmisión ya establecida por la red, y los datos es envía al dispositivo Arduino nano Atmega328 para luego ser procesado de descomprimido los datos.

➤ **Arduino Nano**

El arduino nano recibe por su pin rx la señal digital del dispositivo xbee, la información de datos comprimidos, en dispositivo arduino será descomprimido los datos que será almacenada en una modulo sd-card, además estará encargada de convertir la señal digital en analógica y de enviar la información almacenada hacia el filtro como se observa en la figura 30.



*Figura 30. Arduino nano y cd card
Fuente: Autor*

➤ **Filtro**

El circuito diseñado es un filtro pasa banda, donde la frecuencia de corte inicial es de 16.9 kHz y la frecuencia de corte final es de 17 kHz obteniendo un ancho de banda de 16.90 kHz de una señal y atenúa el paso del resto, además se observa las fórmulas para calcular las resistencias y en la figura 31 se encuentra estructurado el filtro.

Formula

$$F_c = \frac{1}{2\pi R C}$$

Frecuencia de corte inicial

$$F_{c1} = \frac{1}{2\pi R C} = \frac{1}{2\pi \times 270 \times 33n} = 17 \text{kHz}$$

$$F_{c1} = 17 \text{ kHz}$$

Frecuencia de corte final

$$F_{c2} = \frac{1}{2\pi R C}$$

$$F_c = \frac{1}{2\pi \times 150 \times 10u} = 106 \text{ Hz}$$

$$F_{c2} = 106 \text{ Hz}$$

Ancho de banda

$$A_b = F_{c1} - F_{c2}$$

$$A_b = 16.9 \text{ kHz}$$

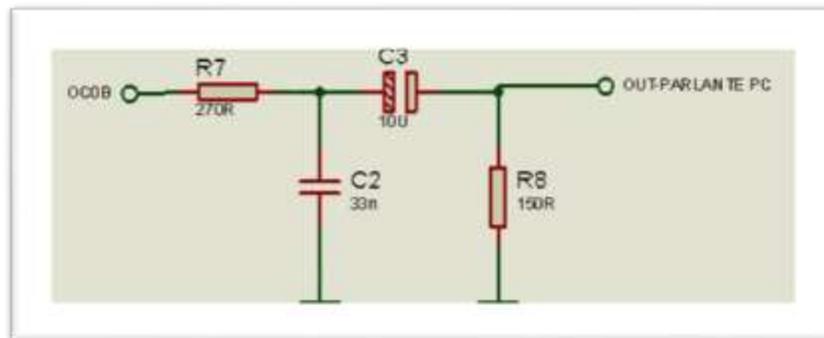


Figura 31. Filtro Pasa Banda

Fuente: Autor

En el siguiente circuito electrónico en donde se encuentra estructurado todos sus elementos que se puede observar en la figura 32. Que fue realizada en el software de simulación de circuito Proteus, que nos ayuda con la simulación de los elementos para realizar las pruebas de todo el circuito receptor.

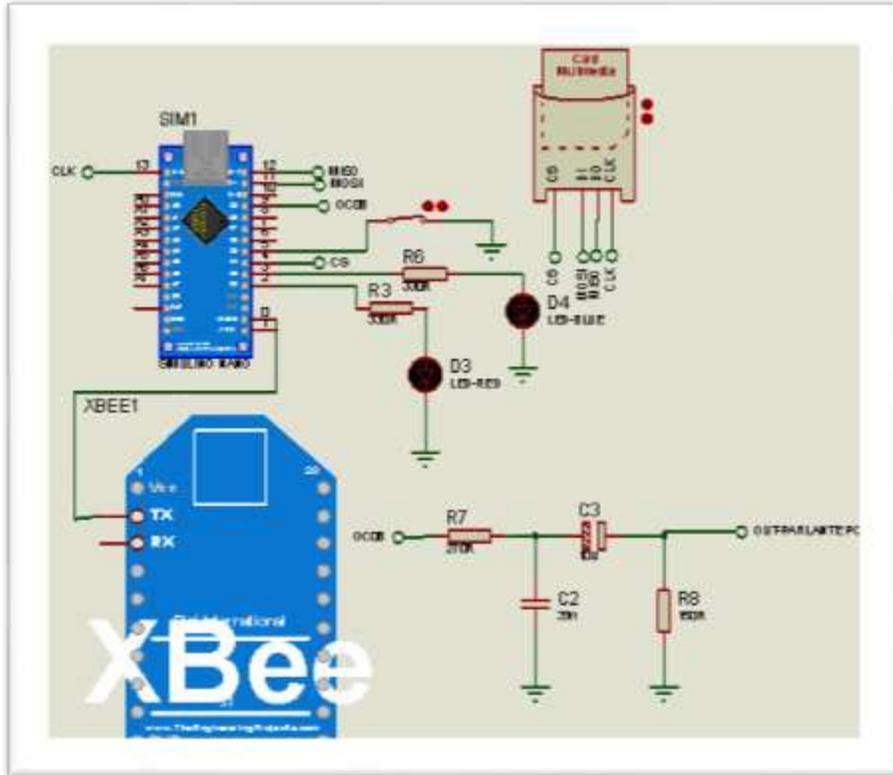
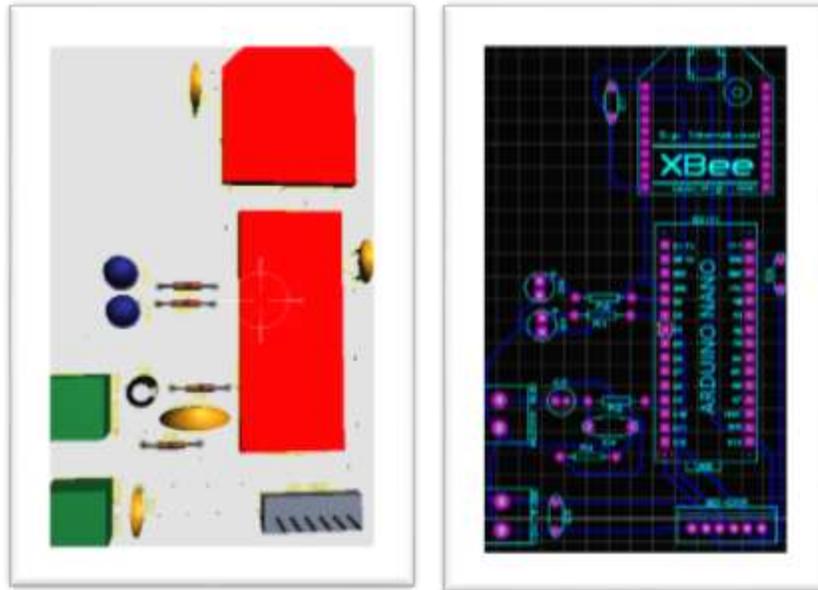


Figura 32. Circuito Receptor
Fuente: Autor

➤ **Diseño de las placas del prototipo**

Para realizar el prototipo del proyecto fue necesario la elaboración de todas las placas y por ende se utilizó el software de simulación Proteus, como ultima placa es la del circuito receptor que fue realizada con las dimensiones de todos los elementos, para ya realizar la placa y visualizar en 3D y con todas las conexiones como se puede apreciar en las figuras 33.



*Figura 33. Diseño de circuito receptor en ARES -Software Proteus
Fuente: Autor*

➤ Descripción del código de Programación

La programación fue realizada en dispositivo arduino Atmega 328p, utilizando las librerías de dispositivo nano V3y que se describe en el diagrama de flujo de la figura 34, del circuito receptor y para mayor análisis se adjunta en el anexo 5. En donde se encuentra la programación detallada de todo los códigos y comandos que se utilizó de todo el circuito receptor para que los datos sean decodificados y transformados en señal de audio, como se observa figura 34 el diagrama de programación de todo el sistema.

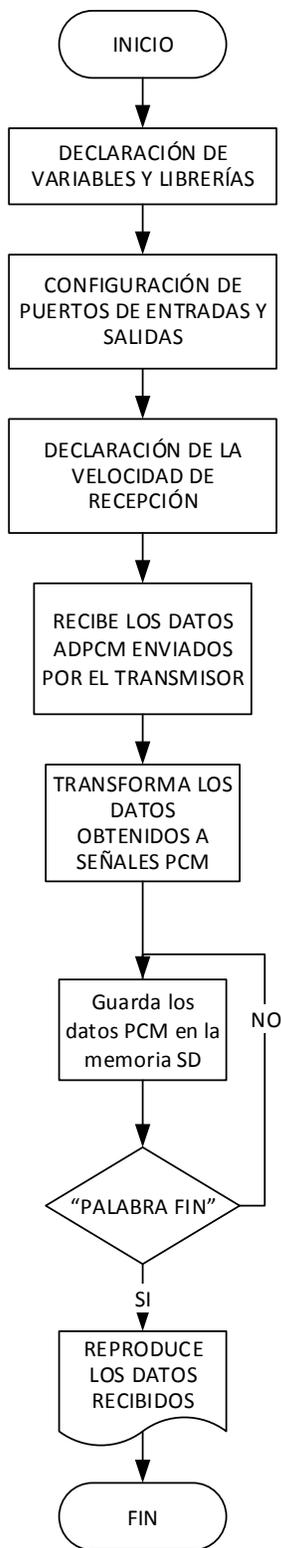


Figura 34. Diagrama de flujo del Receptor
Fuente: Autor

2.10. PROGRAMACIÓN DE DISPOSITIVOS INALÁMBRICOS XBEE

2.10.1. Configuración de la red xbee serie 1

Por primer punto es la instalación de los drivers xbee Explorer USB. El cual nos permite acceder al dispositivo para poder configurar desde la Pc. En la configuración de los xbee S1 como segundo parte, se instala el software XCTU que se puede descargar de la página oficial de xbee, en el programa se selecciona la opción buscar dispositivo xbee.

Ya en el programa se selecciona el puerto COM que se asigna al xbee como se observa en la figura 35.

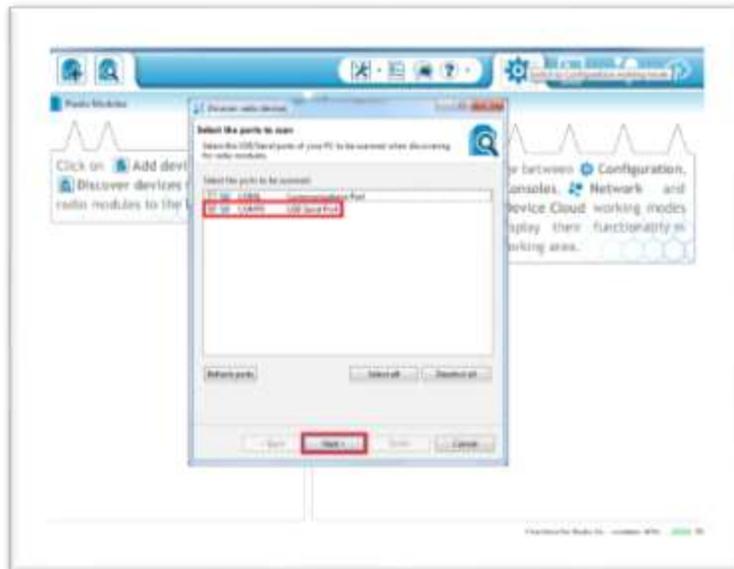


Figura 35. Configuración del dispositivo Xbee S1

Fuente: Autor

Ingresado todos los parámetros en el software para la lectura y configuración del dispositivo Xbee se comienza con la configuración que ya fue establecido para formar la red que nos ayuda para el proyecto y que será realizado en todo dispositivo que forman la red como se observa en la figura 36.

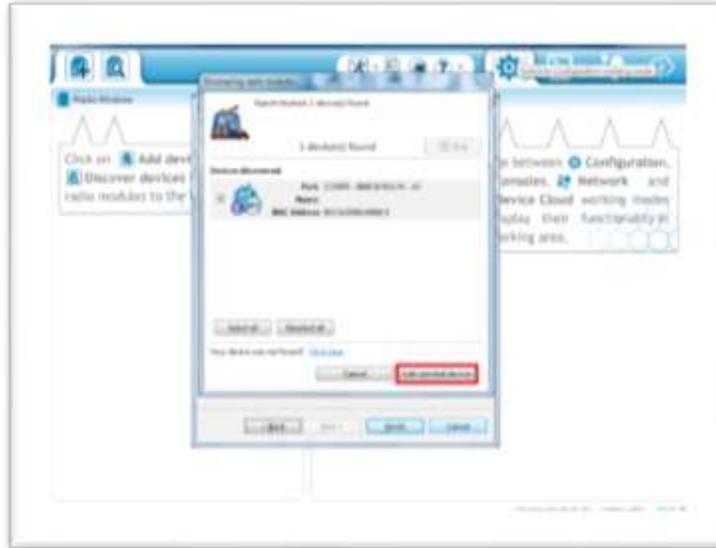


Figura 36. Configuración de los modula xbee S1
Fuente: Autor

En la siguiente fase se modificó los parámetros requeridos como son la velocidad de transmisión que está en 56700 baudios, el canal de transmisión y otros parámetros que fueron analizados para crear la red para el proyecto, eso se configuro a cada dispositivo como el Coordinador en el primer módulo, en el segundo módulo el Rúter y destino final que se requiere para formar la red, como observa en la figura 37.

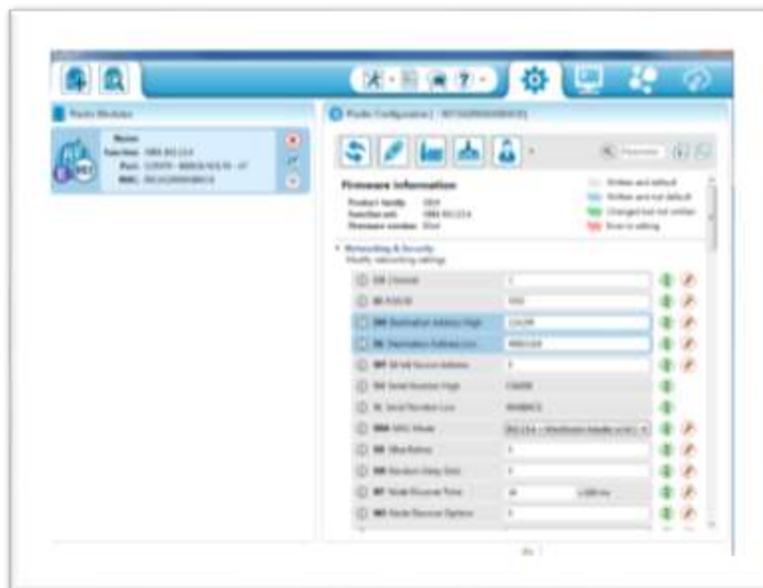


Figura 37. Configuración de los módulos Xbee S1
Fuente: Autor

Con las configuraciones necesarias en los dispositivos Xbee S1, se guarda correctamente todos los parámetros ya establecidos anteriormente.

- Realizar pruebas del de configuración de la red.

En primer lugar, se ubica los dispositivos xbee a una distancia máxima para poder enviar los datos del coordinador hacia el Router y reenviarlo hasta el destino final en donde se comprobará la calidad de la voz y el tiempo estimado que toma en llegar cierta cantidad de datos.

CAPITULO III

3. RESULTADOS

3.1. RESULTADOS DE LAS PRUEBAS REALIZADAS EN LA TRANSMISIÓN DE DATOS CON LOS DISPOSITIVOS INALÁMBRICOS.

- **TIEMPO DE TRANSMISIÓN**

En la siguiente figura38, se obtiene los resultados de varias pruebas realizadas en una red xbee en una distancia de 60 metros con diferentes tamaños de tiempo audio de 5 segundos, 10 segundos y 20 segundos. El parámetro que se tomó en cuenta para realizar el análisis de las pruebas es la efectividad de transmisión de cada tiempo de audio, que depende del tiempo de envío de datos por la red xbee, que a mayor tiempo de transmisión es propenso a que la información llegue incompleta o no llegue los datos y con este parámetro se realizó las pruebas en donde el margen aceptable de una transmisión es por los 90% (Ballesteros, Melo, & Maya Quintero, 2010).

Se realizó los cálculos mediante la siguiente formula:

Audio Enviado 100%

Audio Recibido X

$$X = \frac{\text{Audio Recibido} * 100}{\text{Audio Enviado}}$$

X= Tiempo de reproducción.

Esta fórmula fue aplicada a los diferentes tiempos de audio en todas las pruebas realizadas.

Se muestra que a 20 segundos de audio la tasa de efectividad es del 84%, y en 10 segundos de audio la tasa de efectividad d es del 87%, y a 5 segundos de audio la tasa de efectividad del 89%.

Se determina que a menor tiempo de reproducción es más efectiva la transmisión, que a otros tiempos de reproducción.

EFFECTIVIDAD DE REPRODUCCIÓN A DISTINTOS TIEMPOS

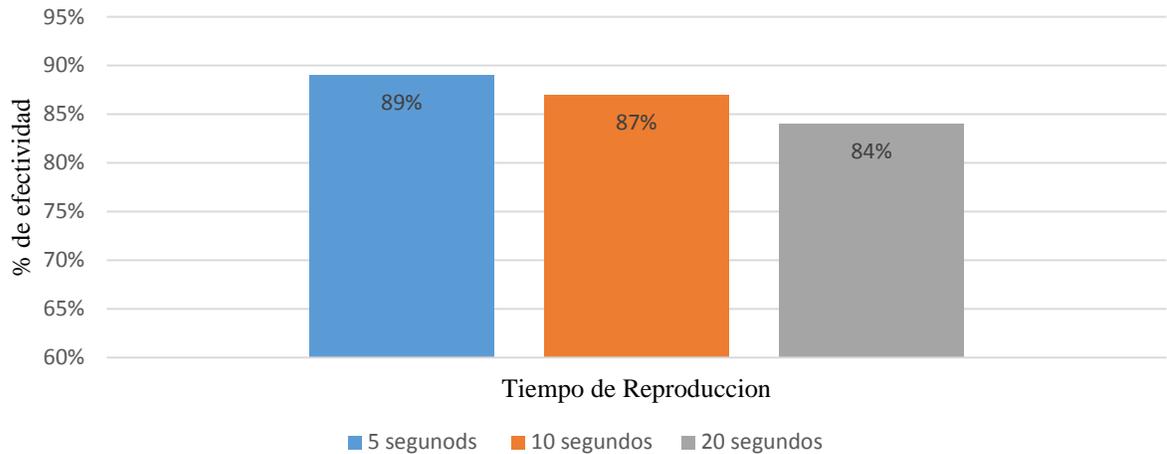


Figura 38. Efectiva de Reproducción
Fuente: Autor

3.2. Pruebas realizadas de la potencia de los xbee s1

La prueba fue realizada a varias distancias y verificadas en la pantalla del analizador de espectros en donde se midió la Potencia en dBm a cada metro con datos transmitidos.

Como se observa en la figura 39 y 40, la frecuencia en que se encuentra, el espectro de la potencia que es señalada con una numeración de 2 para verificar el valor en el cuadro.

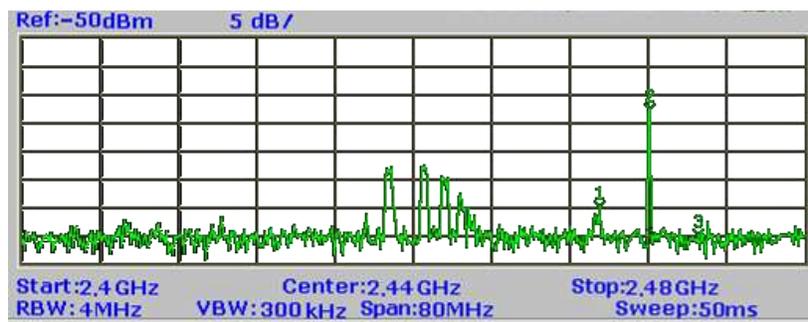


Figura 39. Potencia de la señal
Fuente. Autor

No.	MHz	dBm
1	2459.239	-64.1
2	2463.719	-50.3
3	2469.959	-61.3
4		
5		

Potencia en dBm de la señal

Figura 40. Toma de Datos dBm
Fuente. Autor

En la figura 41 se observa los datos obtenidos anteriormente de la potencia de transmisión de los xbee. De cero a cinco metros la caída de la señal es aproximadamente dos dBm por metro y de 5 a 30 metros es de 0.5 dBm por metro aproximadamente en una condición casi lineal.

RESULTADOS DE LOS DATOS OBTENIDOS DE LA PRUEBAS

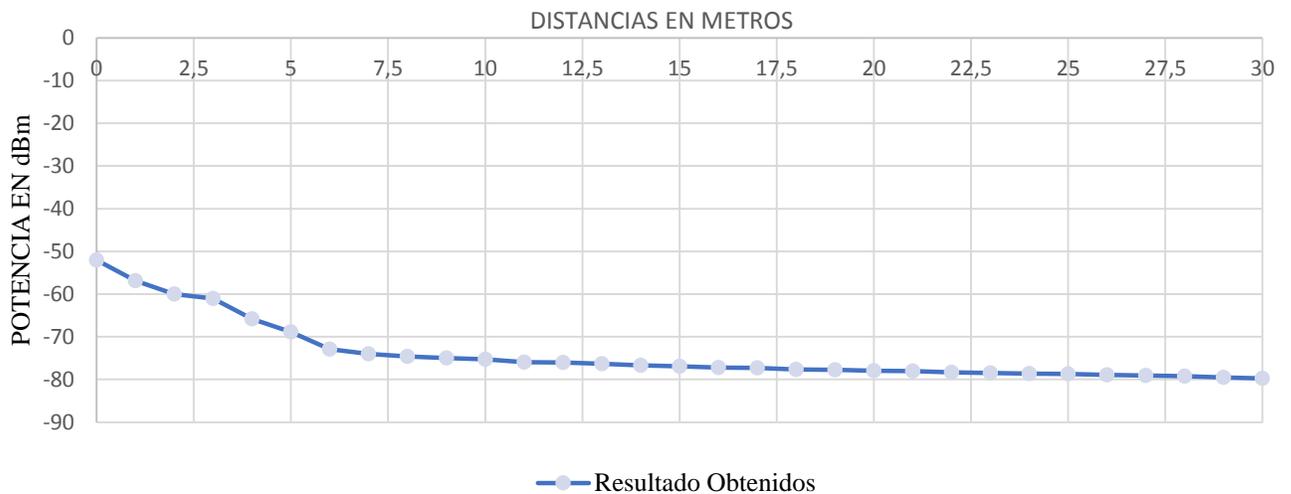


Figura 41. Valores obtenidos a varias distancias
Fuente. – Autor

3.3. COMPROBACIÓN DE LA HIPÓTESIS

Para la comprobar la hipótesis se utiliza el método estadístico Chi-Cuadrado, que es una distribución probabilística continua, que se apoya en un parámetro que representa los grados de libertad, que permite determinar la relación entre dos variables, la H_i alternativa es la que se quiere comprobar y H_o nula (rechaza la hipótesis alternativa) para constatar en cumplimiento de la hipótesis.

H_o = No influye la distancia en una red xbee para la transmisión de datos.

H_i =Si influye la distancia en una red xbee para la transmisión de datos.

3.3.1. Determinación de los valores estadísticos.

Para aceptar o rechazar la hipótesis se toma en cuenta dos escenarios, el primero caso se toma de acuerdo a la distancia que si influye en la transmisión. El segundo escenario que la distancia no influye en la transmisión en una red xbee.

En la tabla 3 se observa los resultados de las pruebas realizados conforman la siguiente tabla, en donde se calcula el total de las filas y las columnas, las cuales sumadas dan un valor total de muestras 60.

Tabla 3. - Datos obtenidos de las pruebas

Método X^2				
METROS	DEFICIENTE	REGULAR	BUENO	TOTAL
55	3	8	9	20
57	8	9	3	20
60	10	8	2	20
Frec. Marginal de columna	21	25	14	60

Fuente. – Autor

Para encontrar la frecuencia teórica esperada de los datos, se calculan mediante la siguiente fórmula

$$F_t = \frac{(T.C) * (T.F)}{T.M}$$

En donde:

F_t = Frecuencia Teórica

T.C = Total de columnas

T.F = Total de filas

De manera que con la fórmula se obtiene la frecuencia teórica que es necesario para calcular el chi cuadrado.

Tabla 4. Frecuencias Teórica calculado

Método X²			
METROS	DEFICIENTE	REGULAR	BUENO
55	7	8.3333	4.66666667
57	7	8.3333	4.66666667
60	7	8.3333	4.6666667

Fuente. – Autor

Con los datos de la tabla, se procede a calcular el Chi cuadrado, que utiliza la siguiente fórmula.

$$X^2 = \sum \frac{(O - E)^2}{E}$$

Donde O = Sumatoria de las frecuencias.

E = Es la frecuencia teórica esperada.

Tabla 5.- Valores del Chi cuadrado

Método X²		
$X^2 = \sum \frac{(O - E)^2}{E}$		
2.28571429	0.01333333	4.02380952
0.14285714	0.05333333	0.5952381
1.28571429	0.01333333	1.52380952

Fuente. - Autor

Con la sumatoria de todos los valores de la tabla se obtiene el valor de Chi cuadrado $X^2 = 9.93714286$, por ende, se hace el análisis de los datos calculados y los datos de la tabla de distribución Chi cuadrado.

Para verificar cuál de las hipótesis es la adecuada se calcula los grados de libertad, que en este caso será: 4 grados de libertad.

Tabla 6. Grados de libertad y probabilidad

GRADOS DE LIBERTAD	PROBABILIDADES			
	0.9	0.5	0.05	0.01
1	0.02	0.46	3.84	6.64
2	0.21	1.39	5.99	9.21
3	0.58	2.37	7.82	11.34
4	1.06	3.36	9.49	13.28

Fuente: Autor

Para la obtención de datos de la tabla se observa el grado de libertad con el margen de error que en este caso es de 0.05, de manera que se obtiene el Chi cuadrado y el valor es 9,4877. En este caso para aceptar la Hipótesis se realizó la comparación.

Si el Chi cuadrado calculado $>$ Chi cuadrado tabla se rechaza hipótesis nula H_0 , Pero si el Chi cuadrado calculado $<$ Chi cuadrado tabla se rechaza hipótesis alternativa H_1 .

De los valores calculados $9,93 > 9,4877$ datos de la tabla, que por la respuesta obtenida se rechaza la hipótesis nula y asigna la hipótesis Alternativa y que en un rango de 55 a 60m la distancia SI influye en la calidad de transmisión de datos.

CAPITULO IV

4. DISCUSIÓN

Los algoritmos de compresión de audio son empleados en muchos campos como en la comunicación, pero para transmitir sobre una red inalámbrica se debe tener en cuenta el tipo de algoritmo que se utiliza, en este proyecto se analiza varios algoritmos que a su vez nos ayuda con bases para estudios más profundos con otros métodos y dispositivos que permitan un mejor desarrollo en compresión y transmisión de datos.

El uso de los algoritmos de compresión va a depender también del tipo de sonido o fuente que se utiliza para trabajar, ya que no da lo mismo comprimir una voz que un audio de música, que en particular es por la cantidad de datos que tiene cada archivo y por la calidad que tiene y utiliza diferentes niveles de compresión para ser transformados en datos y enviados en una red inalámbrica que para tal cometido se utilizó dispositivos didácticos y de bajos recursos.

Con la obtención del proyecto nos damos cuenta que puede transmitir voz por medio de dispositivos inalámbricos de escasa capacidad de memoria y que a su vez cumple con la función que los dispositivos profesionales, pero debe tener en cuenta los parámetros del medio en que se esté transmitiendo para tener mayor efectividad en la transmisión y recepción de voz.

CAPITULO V

5. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Una vez terminada la investigación se concluyó los siguientes:

- Para la evaluación de algoritmos de compresión que se utilizó en el proyecto para su transmisión usando redes Xbee, se recopiló suficiente información y conocimiento con respecto al tema, la cual puede ser muy útil al momento de buscar alternativa en otros algoritmos que funcionen en redes inalámbricas.
- Es posible la transmisión y recepción de voz en redes con dispositivos xbee, no obstante, el dispositivo no fue diseñado para aplicaciones multimedia, pero en situaciones de emergencias donde no se requiera calidad de audio solo utiliza.
- Para lograr una recepción mejor en la calidad de voz, es necesario implementar amplificadores a la salida de los módulos programable Atmega que ahí realiza la conversión señal digital a analógica de los datos y en donde los filtros actúan en la calidad de la voz que es escuchada en los altavoces.
- En la red de dispositivos xbee la potencia va decayendo de acuerdo a la separación de los dispositivos hasta llegar a la distancia máxima y la potencia de la señal se atenúa.

5.2. Recomendaciones

Una vez terminado las conclusiones del proyecto, se propone las siguientes recomendaciones:

- Para mayor recepción de datos en una distancia pre establecido es recomendable colocar los dispositivos Xbee con línea de vista ya que esto nos ayuda a tener mayor recepción de datos.
- Para la alimentación de todo el sistema colocar una batería con una corriente superior al normal para que su funcionamiento sea óptimo.
- Para utilización del algoritmo ADPCM es necesario utilizar dispositivos de una gama alta para no tener inconvenientes en su compleja programación.
- Utilizar los dispositivos xbee en un ambiente sin contaminación de señales de redes para mayor aprovechamiento de la conectividad y transmisión.
- Al prototipo no usar un audio con una calidad muy levada solo la voz y con tiempo de 10 segundos para su mejor transmisión.

CAPITULO VI

6. PROPUESTA

6.1. Título de la propuesta

OPTIMIZACIÓN DE LAS TÉCNICAS DE COMPRESIÓN DE AUDIO SOBRE UNA RED ZIGBEE EN UNA PLATAFORMA DE WSN

6.2. Introducción

En la actualidad gracias al avance de la tecnología se ha podido desarrollar algunos códecs de compresión utilizados para transmitir sonido, que generalmente poseen algoritmos basados en música y en voz, pero muchos de ellos no serían útiles transmitiendo sonidos.

Para la transición de audio es importante tener en cuenta cómo se van a comprimir los datos o que tipos de algoritmos será eficiente en su cometido y así transmitir. El propósito de este proyecto de investigación es realizar un análisis de algoritmo de compresión de audio, para determinar cuál es el más apropiado en cuanto a la transmisión de sonidos en tiempo real. Además, se ha realizado un estudio que abarca la determinación de características técnicas de los diferentes sonidos, con el afán de determinar los rangos de frecuencias en que se encuentran localizados estos sonidos y su ancho de banda dentro del espectro de frecuencia, se analizaron varios códecs de audio utilizados actualmente en transmisión de tiempo real, para poder escoger el más adecuado en este tipo de transición, evitando elegir algún códec que pueda significar un desperdicio de ancho de banda. Después de seleccionar el códec de compresión, se procedió a analizarlo para tener especificaciones de lo que significa utilizar este códec y poder transmitir por medios sensores alámbrico Zigbee.

6.3. Objetivos

6.3.1. Objetivos Generales

Diseñar e implementar un sistema compuesto de técnicas de transmisión de audio para poder transmitir por el limitado ancho de banda de una red inalámbrica Zigbee

6.3.2. Objetivos Específicos

- Aplicar un algoritmo que nos permita comprimir audio con una mínima pérdida de datos para su recepción.
- Diseñar una red de sensores inalámbricos con dispositivos con características aptas para el proyecto que nos ayude a transmitir y recibir audio.
- Comprobar la tasa de transmisión de los dispositivos inalámbricos xbee transmitiendo datos comprimidos de voz.

6.4. Fundamentación científico – técnica

El proyecto puede desarrollarse de una excelente manera en entornos abiertos y despejados para su mejor transmisión y recepción, gracias a un excelente diseño e implementación, además corregir o mejorar la señal de audio con los tipos de compresión que utiliza.

Con la ayuda de algoritmos de compresión de voz, se podrán determinar que algoritmo de compresión es el más efectivo para lograr una comunicación más efectiva y con una mínima cantidad de pérdidas en una red Xbee.

6.5. Descripción de la propuesta

Este proyecto tiene como propósito analizar varios tipos de algoritmo de compresión de voz y utilizar el más indicado que permitan la transmisión y recepción de datos con una pérdida mínima que nos ayude con una calidad de audio aceptable.

Con los módulos Arduino se realiza la conversión analógica/digital y la compresión ADPCM que luego es almacenado en un dispositivo Sd-car que luego es aviado por los módulos Xbee que a su vez se encuentren configurado su protocolo y a una velocidad de transmisión adecuada para la recepción y reproducción de la voz transmitida.

El módulo Xbee ya configurado previamente como destino final para la recepción de datos que a su vez es guardado los datos entrantes en dispositivo Sd-car que pasa x el módulo Arduino para su descompresión y la conversión de Digital a Analógica que luego es conducida a los altavoces para apreciar la calidad de la voz.

6.6. Diseño Organizacional

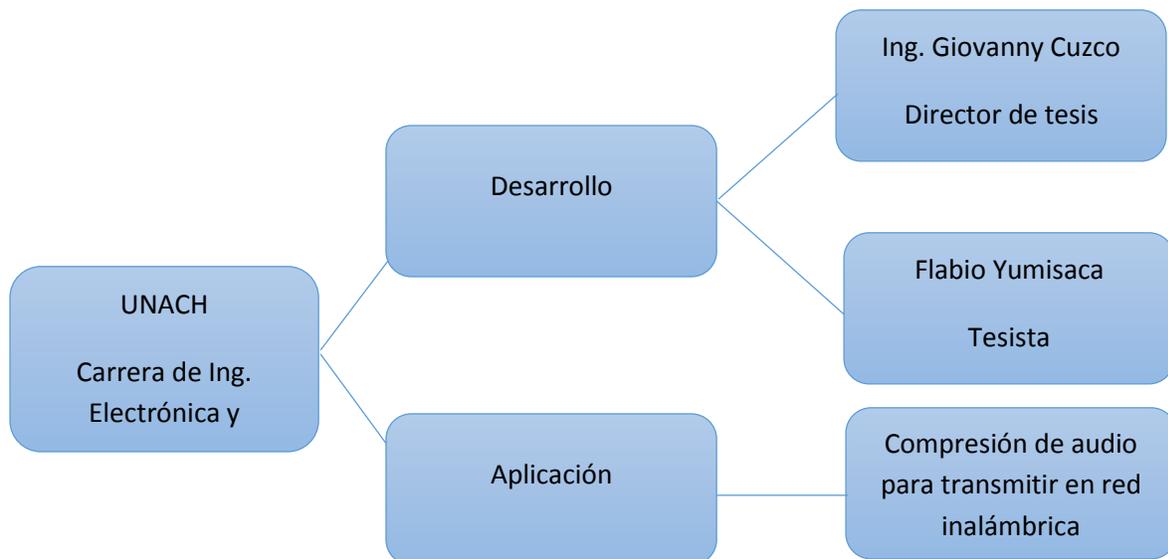


Figura 42. Diseño Organizacional para el desarrollo del proyecto
Fuente. - Autor

6.7. Monitoreo y evaluación del proyecto

La evaluación de la propuesta se realizará sometiendo al sistema a varias pruebas con algoritmo de compresión y elegir el adecuado para proyecto.

El impacto que produce el desarrollo del sistema de compresión de audio es de gran impacto al lograr el envío de datos comprimidos por una red de sensores inalámbricos a varias distancias.

6.8. Cronograma

Tabla 7. Cronograma

ACTIVIDAD	MES 1	MES 3	MES 6	MES 8	MES 10	MES 12
ANÁLISIS Y DISEÑO						
Análisis de los algoritmos de compresión	■					
Seleccionar del algoritmo de compresión por sus características.		■				
Programación en los dispositivos Atmega, el algoritmo seleccionado.			■			
Configuración del protocolo de comunicación en los dispositivos inalámbricos.				■		
Pruebas de la red en las tarjetas de inalámbricas.				■		
IMPLEMENTACION						
Implementación de la red con los dispositivos				■		

7. BIBLIOGRAFÍA

- Aaron, P. (2010). Adaptive Differential Pulse Code Modulation. *Texas Instruments*, 4-7. Obtenido de <https://www.ti.com/lit/an/spra118/spra118.pdf>
- Adarve, J. (25 de 02 de 2015). *Redes Inalámbricas de Malla*. Obtenido de Redes Inalámbricas de Malla: <http://www.blogandvolts.com/2015/02/wireless-mesh-networking.html>
- Arellano, R. (05 de 12 de 2014). *El espectro de frecuencias*. Obtenido de <http://asdopen.unmsm.edu.pe/files/Revista6-1.pdf>
- Ballesteros, D., Melo, H., & Maya Quintero, A. (2010). Sistema de transmisión inalámbrica de señales. *Ingeniería Biomédica*, 55-63.
- Barneda Faudot, I. (2010). Zigbee aplicado a la transmisión de datos. *Etse*, 12-16.
- Barzola, D., Cabrera, R., & Chávez, P. (2014). Comparación entre compresión de audio en diferentes formatos de imágenes. *Compresión de audio*, 1-3.
- Castellanos Rodríguez, A., Padrón Quindemil, F., & Martínez Suárez, F. (2015). Diseño de un sistema electrónico utilizando módulos XBee. *Ingeniería Electrónica, Automática y Comunicaciones*, 5-12.
- Cisco. (18 de 03 de 2012). *Calidad de voz*. Obtenido de https://www.cisco.com/c/es_mx/support/docs/collaboration-endpoints/spa901-1-line-ip-phone/108736-pqa-108736.pdf
- Delgado León, D. (2012). Transmisión de audio usando redes. *Centro de Investigaciones en Microelectrónica*, 12-43.
- Durán Acevedo, C., & García Sierra, H. (2013). Desarrollo de un Sistema Inalámbrico para la Supervisión y Control de un Aerogenerador. *Tecno Lógicas*, 395-409.
- Gonzales, P. N. (30 de Octubre de 2015). *Conversión y Edición de Audio*. Obtenido de <http://jose Luisuribeparedes.blogspot.com/>
- Guzman, J. P. (23 de Mayo de 2012). *protocolo zigbee*. Obtenido de <https://sx-dex.wikispaces.com/ZIGBEE>
- Huidobro, J. M. (2014). Sonido digital y formatos de compresión. *Audio digital*, 10-15.
- Jeff. (16 de 10 de 2008). *Creative Commons*. Obtenido de <https://es.ccm.net/contents/714-la-compresion-de-datos>
- LÓPEZ RODRÍGUEZ, G. J. (02 de 12 de 2014). *Evaluación de algoritmos de compresión de audio*. Obtenido de algoritmos de compresión : <http://dspace.espoeh.edu.ec/handle/123456789/3566>

- Niels, A., & Jan, E. (2006). Redes de sensores inalámbricos. *ABB*, 39-42.
- Pilataxi Duque, B. P. (12 de Noviembre de 2014). Simulacion de código Mp3 en el software Matlab. *Matlab*, págs. 12-13. Obtenido de Simulacion de código Mp3 en el software Matlab .
- Rodriguez Navarro, I. (9 de Abril de 2012). *Desarrollo Multimedia*. Obtenido de Desarrollo Multimedia: <http://www.desarrollomultimedia.es/articulos/tipos-de-formatos-o-archivos-de-audio-y-codecs.html>
- Sanchez, M. (29 de 10 de 2015). *Redes de sensores Inalambricos*. Obtenido de <https://migueysanchez.wordpress.com/2015/09/29/redes-de-sensores-inalambricos-sector-cuatrnario/>
- Tinoco, L. E. (4 de Julio de 2015). *Diferencia entre disco de vinilo, mp3, cd y audio de alta resolución*. Obtenido de <http://highresolutionaudio.blogspot.com/>
- Vialfa, C. (18 de 02 de 2015). *Introduccion al formato Ogg*. Obtenido de <http://es.ccm.net/contents/49-ogg-vorbis>

8. ANEXOS

ANEXO 1 ALGORITMO CODIFICADOR ADPCM

1. ADPCM Encoder

Toma un número firmado de 16 bits
(muestra de habla, 32767 a -32768)
y devuelve un Número de 8 bits que
contiene el código ADPCM de 4 bits (0-15)

Muestra larga

2. Restaurar los valores previos de la
muestra predicha (sp) y el índice de
tamaño de paso del cuantificador

```
predsample = state.prevsample;  
index = state.previndex;
```

3. Encuentre el tamaño de paso del
cuantificador (q) desde una tabla
usando el índice de tamaño de
paso del cuantificador.

```
step = StepSizeTable[index];
```

4. Calcule la diferencia (d) entre
la real muestra (si) y la muestra
predicha (sp).

```
diff = sample - predsamples;
```

5. Establezca el bit de signo del
código ADPCM (t), si es necesario,
y encuentra el valor absoluto
de la diferencia (d).

```
if(diff >= 0)  
code = 0;  
else  
{  
code = 8;  
diff = - diff;
```

6. Guarde el tamaño de paso del
cuantificador (q) de forma
temporal variable.

```
tempstep = step;
```

7. Cuantifique la diferencia (d) en el
código ADPCM (t) usando
el tamaño del paso del cuantificador (q).

```
if(diff >= tempstep)  
{  
code |= 4;  
diff -= tempstep;  
}  
tempstep >>= 1;
```

```

if(diff >= tempstep)
{
    code |= 2;
    diff -= tempstep;
}
tempstep >>= 1;
if(diff >= tempstep)
    code |= 1;

```

9. Cuantice inversamente el código ADPCM (t) en una diferencia predicha (dq) usando el paso cuantificador tamaño (q).

```

diffq = step >> 3;
if(code & 4)
    diffq += step;
if(code & 2)
    diffq += step >> 1;
if(code & 1)
    diffq += step >> 2;

```

10. Verifique el desbordamiento de la nueva muestra pronosticada (sr) sr, que es una muestra firmada de 16 bits, debe estar en el rango de 32767 a -32768.

```

if(predsample > 32767)
    predsampl = 32767;
else if(predsample < -32768)
    predsampl = -32768;

```

11. Encuentre el nuevo índice de tamaño de paso del cuantificador (q) por agregando el índice anterior y una tabla de búsqueda utilizando el código ADPCM (t).

```

index += IndexTable[code];

```

12. Verifique el desbordamiento del nuevo tamaño de paso del cuantificador índice.

```

if(index < 0)
    index = 0;
if(index > 88)
    index = 88;

```

13. la nueva muestra predicha (sr) y cuantificador índice de tamaño de paso para la siguiente iteración.

```

state.prevsampl = predsampl;
state.previndex = index;

```

14. Devuelve el código ADPCM (t).

```

return (code & 0x0f);

```

Anexo 2 Diagrama de flujo ADPCM codificador

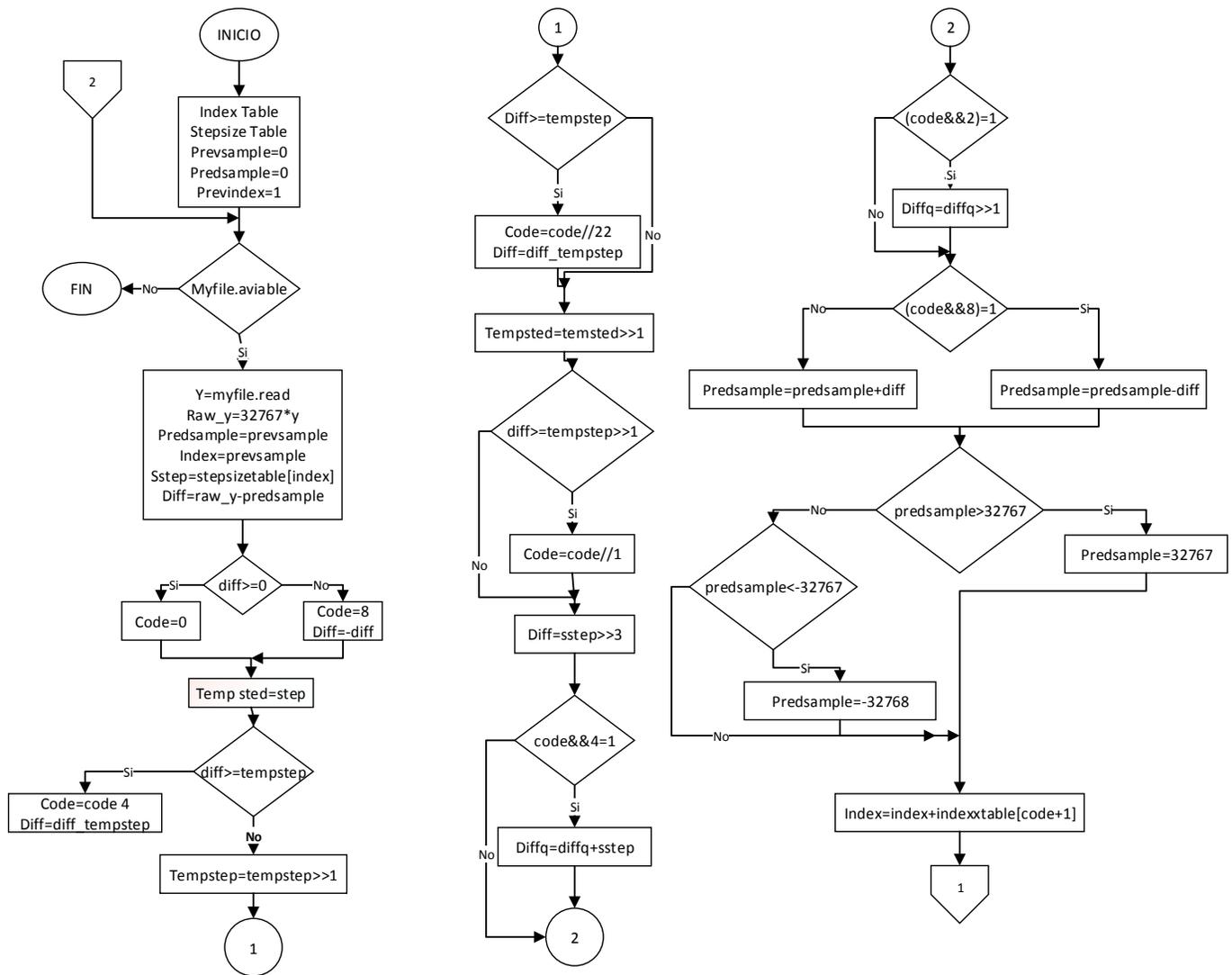


Figura 43. Diagrama de flujo ADPCM
Fuente. Autor

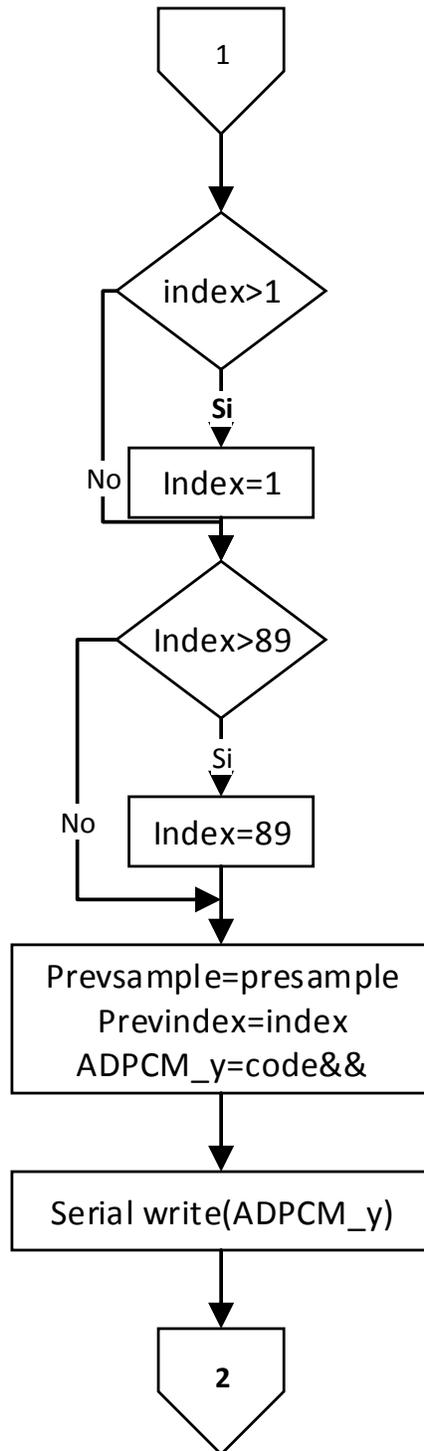


Figura 44. Diagrama de flujo ADPCM
Fuente. Autor

Anexo 3 Diagrama de flujo ADPCM decodificador.

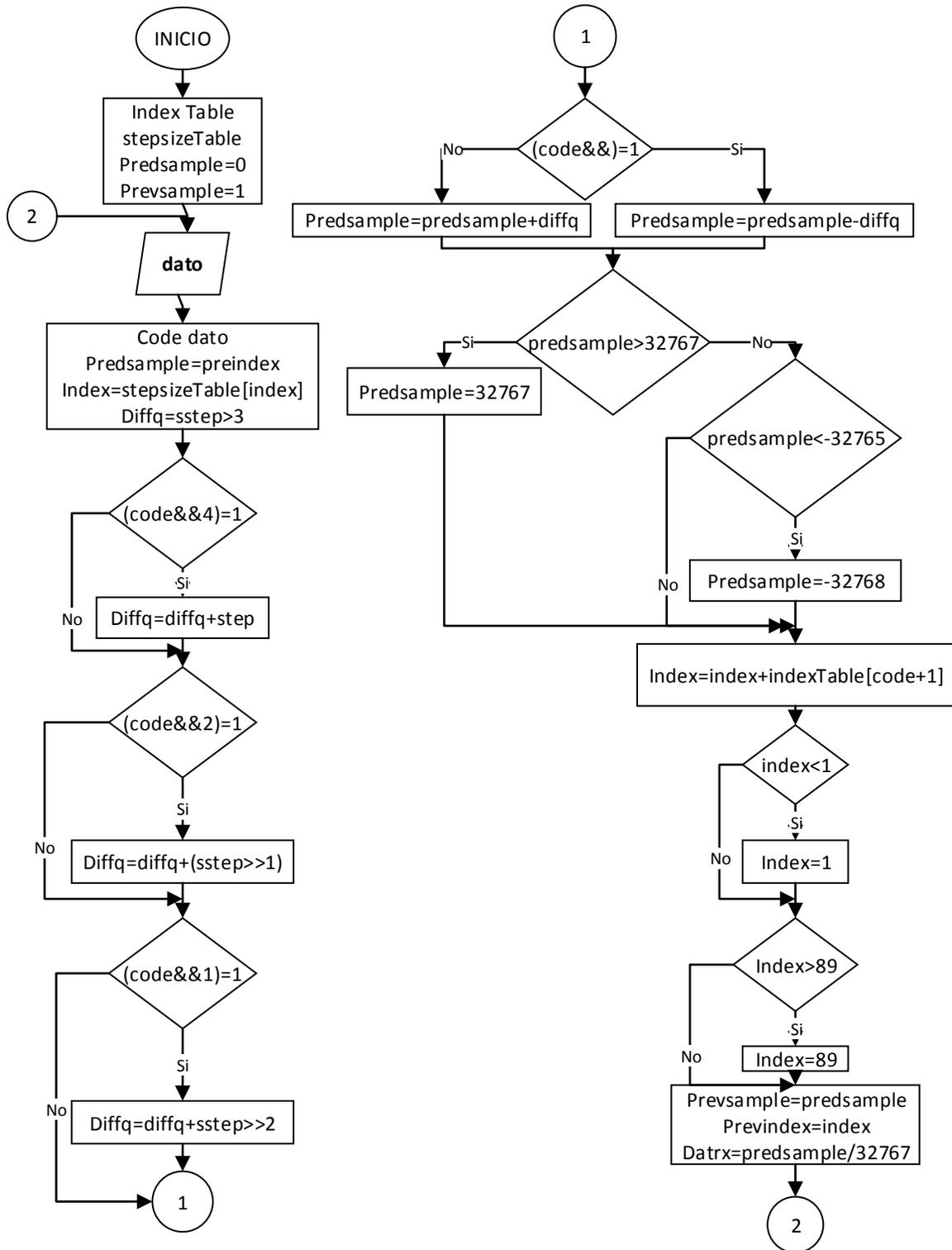


Figura 45. Diagrama de flujo ADPCM Decodificador

Fuente. Auto

Anexo 4 Características de Dispositivo xbee

SPECIFICATIONS	Legacy Digi XBee® S1 802.15.4	Legacy Digi XBee-PRO® S1 802.15.4
PERFORMANCE		
RF DATA RATE	250 kbps	250 kbps
INDOR/URBAN RANGE	100 ft (30 m)	300 ft (100 m)
OUTDOOR/RF LINE-OF-SIGHT RANGE	300 ft (100 m)	1 mi (1.6 km)
TRANSMIT POWER	1 mW (+0 dBm)	60 mW (+18 dBm)*
RECEIVER SENSITIVITY (1% PER)	-92 dBm	-100 dBm
DIGI HARDWARE	S1	
TRANSCEIVER CHIPSET	Freescale MC13212	
FEATURES		
SERIAL DATA INTERFACE	3.3V CMOS UART	
CONFIGURATION METHOD	API or AT Commands, local or over-the-air	
FREQUENCY BAND	2.4 GHz	
INTERFERENCE IMMUNITY	DSSS (Direct Sequence Spread Spectrum)	
SERIAL DATA RATE	1200 bps - 250 kbps	
ADC INPUTS	(6) 10-bit ADC inputs	
DIGITAL I/O	8	
ANTENNA OPTIONS	Chip, Wire Whip, U.FL, & RPSMA	
NETWORKING & SECURITY		
ENCRYPTION	128-bit AES	