



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES

TEMA:

**Gestión de redes de sensores inalámbricos (WSN) para la industria  
petrolera**

AUTOR:

Dender Zurita, Roberto Carlos

Trabajo de Titulación previo la obtención del Grado Académico de  
**MAGÍSTER EN TELECOMUNICACIONES**

TUTOR:

Zamora Cedeño, Néstor Armando

Guayaquil

20 de Octubre del 2017



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por el Magíster **Dender Zurita, Roberto Carlos** como requerimiento parcial para la obtención del Grado Académico de **MAGÍSTER EN TELECOMUNICACIONES.**

TUTOR

---

Zamora Cedeño, Néstor Zamora

DIRECTOR DEL PROGRAMA

---

Romero Paz, Manuel de Jesús

Guayaquil, a los 20 días del mes de Octubre del año 2017



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES

**DECLARACIÓN DE RESPONSABILIDAD**

Yo, **Dender Zurita, Roberto Carlos**

DECLARÓ QUE:

El trabajo de titulación “**Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera**”, previa a la obtención del grado Académico de **Magíster en Telecomunicaciones**, ha sido desarrollado, respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico de la tesis del Grado Académico en mención.

Guayaquil, a los 20 días del mes de Octubre del año 2017

EL AUTOR

---

Roberto Carlos Dender Zurita



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES

**AUTORIZACIÓN**

Yo, **Dender Zurita, Roberto Carlos**

Autorizamos a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución de la Tesis de Maestría titulada: **“Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 20 días del mes de Octubre del año 2017

EL AUTOR

---

Roberto Carlos Dender Zurita

# REPORTE DE URKUND

**URKUND**

**Documento** [Roberto Dender final.docx](#) (D30297804)

**Presentado** 2017-08-30 13:17 (-05:00)

**Presentado por** rd\_yanio@hotmail.com

**Recibido** edwin.palacios.ucsg@analysis.orkund.com

**Mensaje** ROBERTO DENDER TESIS MAESTRIA TELECOMUNICACION [Mostrar el mensaje completo](#)

1% de estas 32 páginas, se componen de texto presente en 1 fuentes.

**Lista de fuentes Bloques**

Categoría	Enlace/nombre de archivo	
	<a href="http://docplayer.es/36323033-Universidad-catolica...">http://docplayer.es/36323033-Universidad-catolica...</a>	<input type="checkbox"/>
	<a href="http://docplayer.es/44778506-Msc-manuel-romero...">http://docplayer.es/44778506-Msc-manuel-romero...</a>	<input type="checkbox"/>
	<a href="http://dspace.ups.edu.ec/handle/123456789/4383">http://dspace.ups.edu.ec/handle/123456789/4383</a>	<input type="checkbox"/>
	<a href="http://quimicoglobal.mx/productos-quimicos-indu...">http://quimicoglobal.mx/productos-quimicos-indu...</a>	<input checked="" type="checkbox"/>
	<a href="https://doi.org/10.1109/ICTP.2015.7427934">https://doi.org/10.1109/ICTP.2015.7427934</a>	<input type="checkbox"/>

0 Advertencias. Reiniciar Exportar Compartir

UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL SISTEMA DE POSGRADO MAESTRÍA EN TELECOMUNICACIONES

TEMA: Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera

AUTOR: Dender Zurita, Roberto Carlos

Trabajo de Titulación previo la obtención del Grado Académico de Magíster en Telecomunicaciones

TUTOR: Zamora Cedeño, Néstor Armando

Guayaquil

31 de Julio del 2017

UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO MAESTRÍA EN TELECOMUNICACIONES

## **Dedicatoria**

Con todo mi cariño y mi amor para las personas que hicieron todo en la vida para que yo pudiera lograr mis sueños, por motivarme y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Roberto Carlos Dender Zurita

## **Agradecimientos**

### **A Dios**

Por la bendición y la oportunidad que él me ha dado para continuar mis estudios superiores.

### **A esposa y a mi hija**

Por haber confiado en mí y por todo el esfuerzo y sacrificio de encaminarme por el camino correcto.

### **A mis padres**

Que han colaborado de diferentes maneras para la consecución de este importante objetivo.

Roberto Carlos Dender Zurita



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES

**TRIBUNAL DE SUSTENTACIÓN**

f. \_\_\_\_\_

**M. Sc. Néstor Zamora Cedeño**  
TUTOR

f. \_\_\_\_\_

**M. Sc. Manuel Romero Paz**  
DIRECTOR DEL PROGRAMA

f. \_\_\_\_\_

**M. Sc. Celso Bohórquez Escobar**  
REVISOR

f. \_\_\_\_\_

**M. Sc. Edwin Palacios Meléndez**  
REVISOR



## ÍNDICE GENERAL

ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABLAS.....	XIV
Resumen.....	XV
Abstract.....	XVI
Capítulo 1: Descripción del proyecto de intervención.....	17
1.1. Introducción.....	17
1.2. Antecedentes.....	18
1.3. Definición del problema.....	20
1.4. Justificación del Problema a Investigar.....	21
1.5. Objetivos.....	22
1.5.1. Objetivo General:.....	22
1.5.2. Objetivos específicos:.....	22
1.6. Hipótesis o Idea a Defender.....	22
1.7. Metodología de investigación.....	22
Capítulo 2: Fundamentos Teóricos.....	24
2.1. Introducción.....	24
2.2. Visión general.....	25
2.3. Red de Sensores Inalámbricos.....	26
2.3.1. Aplicaciones redes de sensores inalámbricos.....	28
2.3.2. Detección de eventos en redes de sensores inalámbricos.....	29
2.4. Calidad de servicio en redes de sensores inalámbricos.....	30
2.5. Estándares de ZigBee.....	31
2.5.1. Protocolo IEEE 802.15.4.....	31
2.5.2. Funcionamiento del ZigBee.....	33
2.5.3. Capas de ZigBee.....	37
2.6. Radio XBee.....	39
2.6.1. Formato de envío y recepción de datos.....	41
2.6.2. Configuración del radio XBee.....	42
2.6.3. Puertos de Entrada/Salida.....	43

2.6.4. Modos de funcionamiento .....	45
2.7. Plataforma Arduino.....	46
2.8. Protocolos inalámbricos apropiados para el ambiente y requerimientos de la industria petrolera. ....	48
2.8.1. Estándar WirelessHART. ....	48
2.8.1.1. Arquitectura WirelessHART. ....	50
2.8.1.2. Capas de protocolos WirelessHART.....	51
2.8.1.3. Componentes de un sistema WirelessHART. ....	52
2.8.2. Estándar ISA SP 100. ....	53
Capítulo 3: Diseño para la implementación de una red de sensores inalámbricos en la industria petrolera.....	55
3.1. Análisis de campo. ....	55
3.2. Elementos utilizados para la red inalámbrica de sensores. ....	56
3.2.1. Módulo ESP8266.....	56
3.2.2. Router D-link DIR-600.....	58
3.2.3. CPU AMD Turbo 3600 i686 .....	59
3.3. Diseño de red inalámbrica.....	59
3.4. Diseño en ISIS Proteus y programación de los sensores ESP8266. ....	60
3.5. Implementación y configuración del servidor web .....	65
Capítulo 4: Conclusiones y Recomendaciones.....	78
4.1. Conclusiones.....	78
4.2. Recomendaciones.....	79
Bibliografía.....	80

## ÍNDICE DE FIGURAS

### Capítulo 2:

Figura 2. 1: Vista general de una red de sensores inalámbricos. ....	24
Figura 2. 2: Elementos básicos de un nodo sensor.....	25
Figura 2. 3: Componentes de un módulo de nodo Sensor. ....	27
Figura 2. 4: Topología estrella de una red ZigBee. ....	33
Figura 2. 5: Topología estrella de una red ZigBee. ....	34
Figura 2. 6: Topología enmallada de una red ZigBee. ....	35
Figura 2. 7: Ejemplo de aplicación para la topología de una red ZigBee.....	36
Figura 2. 8: Capas ZigBee e IEEE 802.15.4. ....	38
Figura 2. 9: Esquemático de la configuración de equipos utilizados para pruebas. ....	40
Figura 2. 10: Tipos de antenas para XBee.....	40
Figura 2. 11: Tabla de datos del modelo de tramas API. ....	41
Figura 2. 12: Modelo la API de identificación 0x83. ....	42
Figura 2. 13: Configuración del módulo de una radio.....	43
Figura 2. 14: Pinza del radio XBee. ....	43
Figura 2. 15: Modos de funcionamiento del XBee.....	46
Figura 2. 16: Placa Arduino Duemilanove. ....	46
Figura 2. 17: Diagrama de una red WirelessHART genérica.....	49
Figura 2. 18: Esquemático de la arquitectura WirelessHART.....	51
Figura 2. 19: Diagrama de las capas de WirelessHART. ....	52
Figura 2. 20: Componentes de una red WirelessHART ....	53

### Capítulo 3:

Figura 3. 1: Tanques de almacenamiento de combustible. ....	55
Figura 3. 2: Campo de tanques de almacenamiento de GLP.....	56
Figura 3. 3: Modulo Wifi ESP8266. ....	57
Figura 3. 4: Pinout ESP8266. ....	58
Figura 3. 5: Router D-Link.....	58
Figura 3. 6: Detalles de CPU utilizado como servidor debian.....	59
Figura 3. 7: Topología utilizada en la implementación de WSN. ....	60

Figura 3. 8: Esquemático para la simulación de sensor de temperatura. ....	61
Figura 3. 9: Declaración de librerías y variables. ....	62
Figura 3. 10: Función para conexión de red inalámbrica wifi.....	63
Figura 3. 11: Estado de conectividad a red inalámbrica wifi.....	63
Figura 3. 12: Función de conectividad a servidor mqtt. ....	64
Figura 3. 13: Función Setup de pines de la placa ESP8266. ....	64
Figura 3. 14: Función principal del código.....	65
Figura 3. 15: Interfaz de Putty.....	66
Figura 3. 16: Ingreso al servidor via SSH.....	66
Figura 3. 17: Procedimiento inicial de instalación del núcleo del sistema de gestión.....	67
Figura 3. 18: Descarga de paquete Domogik desde github.....	68
Figura 3. 19: Descarga de paquete Domogik-mq desde github. ....	68
Figura 3. 20: Comando para instalación de Domogik-mq.....	68
Figura 3. 21: Comando para instalación de Domogik.....	68
Figura 3. 22: Mensaje de confirmación de correcta instalación de paquetes. ....	69
Figura 3. 23: Creación de base de datos. ....	69
Figura 3. 24: Actualización por defecto de domogik.....	70
Figura 3. 25: Ejecución de servidor Mosquitto. ....	70
Figura 3. 26: Instalación de plugin mqtt para sistema de gestión. ....	70
Figura 3. 27: Instalación de paquetes requeridos para interfaz web. ....	71
Figura 3. 28: Inicio de sesión para administrar desde el backend.....	71
Figura 3. 29: Lista de plugin instalados en el sistema. ....	72
Figura 3. 30: Configuración de dispositivos clientes del servidor mqtt. ....	72
Figura 3. 31: Lista de dispositivos para implementación de obtención de resultados.....	73
Figura 3. 32: Seguimiento de datos recibidos al servidor.....	73
Figura 3. 33: Interfaz gráfica web del sistema de gestión. ....	74
Figura 3. 34: Secciones de interfaz web. ....	74
Figura 3. 35: Configuración de secciones y widgets. ....	75
Figura 3. 36: Interfaz gráfica de resultados de sensor de temperatura simulada .....	75
Figura 3. 37: Dato estadístico de las muestras tomadas en días. ....	76

Figura 3. 38: Interfaz gráfica de resultados de sensor de presión. ....	76
Figura 3. 39: Dato estadístico para sensor de presión. ....	77

## ÍNDICE DE TABLAS

### Capítulo 2:

Tabla 2. 1: Rangos de frecuencia del protocolo IEEE 802.15.4. ....	32
Tabla 2. 2: Capa de protocolo ISO 100.11a. ....	54

## Resumen

Para el presente trabajo de titulación se describieron los conceptos sobre la red de sensores inalámbricos, su composición y arquitectura. Así, como también los estándares de comunicación inalámbrico y su implementación sobre el estándar IEEE 802.15. Las redes de sensores inalámbricos (WSN) hoy en día son de gran ayuda por lo que permiten llegar a lugares dificultosos ya sean estos de gran altura o lugares confinados lo que facilita su acceso a la obtención de datos de una manera fácil, confiable y en línea sin necesidad de infraestructuras físicas. Posteriormente, se implementó una red de sensores inalámbricos que permita la expansión del sistema de una manera rápida debido a que existen múltiples aplicaciones para la obtención y procesamiento de datos. Después, se enfocó en la creación de un sistema de gestión de la red inalámbrica de sensores a escala para el modelado de una red de sensores inalámbricos en la industria petrolera. Los resultados obtenidos permitieron obtener mediciones de los eventos ocurridos en cada uno de los sensores que fueron evaluados en la WSN.

**Palabras Claves:** ZIGBEE, SENSORES, REDES, DATOS, INALÁMBRICO, WSN.

## **Abstract**

For the present work of titration the concepts on the network of wireless sensors, their composition and architecture were described. Thus, as well as wireless communication standards and their implementation on the IEEE 802.15 standard. Wireless sensor networks (WSNs) today are of great help, allowing you to reach difficult places, be they high altitude or confined places, which facilitates their access to obtaining data in an easy, reliable and Line without the need for physical infrastructures. Subsequently, a network of wireless sensors was implemented that allows the expansion of the system in a fast way because there are multiple applications for obtaining and processing data. He then focused on the creation of a wireless sensor network management system for the modeling of a wireless sensor network in the oil industry. The results obtained allowed to obtain measurements of the events occurred in each one of the sensors that were evaluated in the WSN.

**KEYWORDS:** Sensors, Network, Data, Wireless, Communication



## **Capítulo 1: Descripción del proyecto de intervención.**

El presente capítulo se describe la perspectiva general del trabajo de titulación relacionada a la red de sensores inalámbricos y cuya problemática de aplicación de la industria petrolera se propone un sistema de gestión de red.

### **1.1. Introducción.**

La industria de petróleo y gas incluye procesos de exploración, extracción, refinación, transporte y comercialización de productos petrolíferos. El mayor volumen de productos de la industria son el combustible, el petróleo y la gasolina (nafta). El petróleo es también la materia prima para muchos productos químicos, incluyendo productos farmacéuticos, solventes, fertilizantes, plaguicidas y plásticos. A medida que la demanda de combustibles fósiles continúa creciendo, las compañías de petróleo y gas tendrán que desarrollar nuevas tecnologías y mejorar las operaciones con el fin de aumentar la productividad y ampliar sus capacidades actuales.

Además, las regulaciones medioambientales están cambiando constantemente y cada vez son más estrictas día a día. En 2008, la Comisión de Conservación de Petróleo (*Oil Conservation Commission, OCC*) aprobó un conjunto de normas sobre la gestión de los residuos superficiales procedentes de las operaciones de petróleo y gas que obligan a las empresas a transportar el suelo y el agua altamente contaminados a sitios de eliminación permanente. La OCC también aprobó sanciones más estrictas para hacer cumplir la industria con las normas ambientales.

Las compañías de petróleo y gas deben desarrollar nuevos métodos para cumplir con las nuevas regulaciones y reducir los accidentes y las emisiones sin afectar la producción. Debido a la evolución de las tecnologías digitales y las comunicaciones inalámbricas, las redes de sensores inalámbricos pueden organizarse rápidamente y adaptarse continuamente

para monitorear y controlar las condiciones ambientales y maquinaria en respuesta a los requerimientos del negocio.

Las tecnologías inalámbricas han evolucionado rápidamente, lo que ha permitido obtener datos y facilitar las comunicaciones, en el año 2003 según el Instituto Tecnológico de Massachusetts (MIT), identificó 10 tecnologías salientes, entre las cuales surge WSN (Wireless Sensor Networks), a partir de este suceso el avance de esta tecnología ha incrementado y se están implementando millones de redes de este tipo a lo largo del mundo. Inicialmente su principal aplicación de este tipo de redes fue en el campo militar, ahora se han desarrollado a muchas aplicaciones para las áreas industriales y de interés social, así como en control de procesos de producción, control de tráfico, monitoreo de la salud y automatización de hogares.

Los recientes avances en la tecnología de detección inalámbrica fomentan la optimización y mejora de los procesos de desarrollo de productos y prestación de servicios. Las Redes de Sensores Inalámbricos Industriales (IWSN) son una clase emergente de WSN que enfrenta restricciones específicas relacionadas con las particularidades de la producción industrial.

En estos términos, IWSN se enfrenta a varios desafíos como la fiabilidad y robustez en entornos agresivos, así como la capacidad de ejecutar correctamente y alcanzar la meta en paralelo con todos los otros procesos industriales. Además, las soluciones de IWSN deben ser versátiles, fáciles de usar e instalar, de larga vida útil y dispositivos de bajo costo - de hecho, la combinación de requisitos difíciles de cumplir.

## **1.2. Antecedentes.**

Durante la búsqueda de información de trabajos relacionados con aplicaciones de redes de sensores inalámbricos en el sector industrial y en beneficio de la sociedad. Las WSN permiten conectar múltiples sensores para

alguna aplicación específica. Por ejemplo, Ecuador no es la excepción y comienza a incursionar en este campo, gracias al proyecto realizado por investigadores de los siguientes centros de educación superior: Universidad de las Fuerzas Armadas “ESPE”, Universidad de Harvard y la Universidad de Carolina del Norte que recientemente implementaron una WSN, la que permite monitorear las erupciones en el Volcán Tungurahua. (Azüero, 2016).

Otra referencia acerca de las redes de sensores inalámbricos es que se brinda la posibilidad de utilizar una nueva tecnología en el área agrícola para optimizar el uso de un recurso no renovable, como el agua, mediante una interfaz amigable con el usuario, de fácil manejo, logrando así fusionar dos áreas que aparentemente no tienen relación (Maya, 2014)

Los autores (Rao, Santhi, Professor, & Lavanya, 2012) en su investigación se centran en el monitoreo y protección de operaciones de petróleo y gas utilizando redes de sensores inalámbricos que están optimizadas para disminuir la instalación, y el costo de mantenimiento, requerimientos de energía, aumentar la confiabilidad y mejorar la eficiencia de la comunicación. Además, de que presentan experimentos de simulación utilizando el modelo propuesto, que permiten proporcionar nuevas herramientas para la investigación en mantenimiento predictivo y monitoreo basado en condiciones de maquinaria de fábrica en general y para sistemas de mecanizado de arquitectura abierta en particular.

La mayor parte de entornos petroleros en nuestro país no tienen un sistema de comunicación inalámbrico para la obtención de los datos y por esto siempre se dependerá del factor humano o de sistemas antiguos para el traslado de la información hacia las centrales, es notorio que las oportunidades que hoy en día brindan las TICs, para la monitorización y control temprana de procesos industriales mediante el uso de WSN, resultaría beneficioso para la sociedad y el país.

La detección inalámbrica mediante sensores ya no necesita ser relegada a lugares donde el acceso es difícil o donde el cableado no es práctico. Los sistemas de monitoreo de condiciones inalámbricas pueden implementarse de manera rentable en aplicaciones extensas que históricamente se manejaban al ejecutar rutas con recolectores de datos.

Mientras que para (Obodoeze, Inyama, & Idigo, 2012) en su investigación, introducen arquitecturas prácticas de implementación y de mecanismos que pueden asegurar que las instalaciones de petróleo y los sensores inalámbricos sean atacados físicamente para que puedan supervisar con éxito y reportar con éxito las incidencias de vandalización de tuberías y equipos de manera fácil y puntual, manteniendo al mismo tiempo la seguridad de los datos de la red de sensores inalámbricos.

### **1.3. Definición del problema**

La tecnología WSN presenta importantes beneficios en costo, facilidad de implementación, flexibilidad y conveniencia en relación a la alternativa cableada que está bien establecida en la industria. Investigaciones significativas apuntan a la proyección del despliegue generalizado de WSN en la automatización industrial. Las WSNs son consideradas dentro de las líneas de investigación de comunicaciones inalámbricas de la Maestría en Telecomunicaciones de la Universidad Católica Santiago de Guayaquil. Aunque no existe un estudio preliminar sobre lo que sería el uso de las tecnologías WSN para el monitoreo de variables de procesos, puesto que para la realización de este trabajo de titulación se enfoca en el área de aplicación de la industria petrolera.

La industria del petróleo y el gas es una industria importante que tiene un tremendo impacto en la economía global. Incluye procesos de exploración, extracción, refinación, transporte y comercialización de productos petrolíferos. En el transcurso de los procesos anteriores, existe la necesidad de un control extensivo de diversos parámetros a través de la ayuda de un gran número de

sensores. Estos sensores se colocan en diferentes lugares para medir diferentes datos sobre el rendimiento de la planta y el entorno operativo.

De acuerdo a lo expuesto, surge la necesidad de desarrollar un sistema de gestión de redes de sensores inalámbricos (WSN) que permita monitorear tuberías y diferentes procesos específicos en el sector petrolero.

#### **1.4. Justificación del Problema a Investigar.**

Con el paso de los años, los sensores desplegados han trabajado eficazmente juntos a través de enlaces de cable por cable. Estos sensores con cable son costosos de desplegar, operar y mantener, no son los más adecuados para instalaciones temporales y son difíciles de configurar en ambientes hostiles y remotos. La tecnología de red inalámbrica de sensores (WSN) proporciona una opción más rápida, menos costosa, más flexible y más conveniente para los sistemas de sensores con cable.

Los avances en el campo de Internet, las comunicaciones y las tecnologías de la información también han contribuido al desarrollo de redes de sensores inalámbricos (WSNs). Las WSNs tienen una aplicación tremenda en la automatización industrial para la fabricación y automatización de control de procesos a través de la supervisión de parámetros como temperatura, presión y flujo.

Las perspectivas de futuro para el impacto de WSN son muy prometedoras. Es decir, que la industria proyecta que habrá cerca de 24 millones de sensores inalámbricos y actuadores desplegados en todo el mundo en 2017 y que la tecnología WSN tendría una mayor influencia en las aplicaciones. Con una consideración más restringida de la industria de petróleo y gas, el impacto de WSN es también muy prometedor como se informa en la literatura.

## **1.5. Objetivos**

### **1.5.1. Objetivo General:**

Diseñar un sistema de gestión de una red de sensores inalámbricos para la industria petrolera.

### **1.5.2. Objetivos específicos:**

- ✓ Describir el estado del arte de las comunicaciones inalámbricas y de redes de sensores inalámbricos.
- ✓ Revisar los estándares y protocolos existentes para la implementación de redes de sensores inalámbricos.
- ✓ Diseñar a escala una red de sensores inalámbricos (WSN) para una estación petrolera.
- ✓ Evaluar la red de sensores inalámbricos e interpretar los resultados para impulsar su instalación.

## **1.6. Hipótesis o Idea a Defender**

En este trabajo, se ha presentado una visión general de la aplicación de WSN en la industria petrolera. La tecnología WSN presenta importantes beneficios en costo, facilidad de implementación, flexibilidad y conveniencia en relación a la alternativa cableada que está bien establecida en la industria. Investigaciones significativas apuntan a la proyección del despliegue generalizado de WSN en la automatización industrial. Sin embargo, la tecnología no está exenta de desafíos únicos que han sido destacados aquí como áreas de investigación.

## **1.7. Metodología de investigación.**

En diferentes escenarios encontrados relacionados a la metodología de investigación científica, se pudo evidenciar, que, para las telecomunicaciones, el ámbito de investigación tiene un corte cuantitativo, y a veces mixto (cuali-cuantitativo). La metodología de investigación utilizada en el presente trabajo de titulación, es del tipo descriptivo y explicativo con enfoque empírico-analítico. Se puede considerar que la simulación y programación es otro

método de investigación, pero propio. Esto nos permite el desarrollo del sistema de gestión de redes de sensores inalámbricos (WSN) para la industria petrolera.

Aunque, existen muchas técnicas de investigación que incluyen métodos estadísticos y de intersección de covarianza (IC), que son utilizados para mejorar la calidad de servicio (QoS) en WSNs. Sin embargo, la mayoría de ellos no son capaces de hacer frente a la incertidumbre de los datos producidos por WSNs. Además, la inflexibilidad de los métodos impide procesar los datos de manera realista.

## Capítulo 2: Fundamentos Teóricos.

Este capítulo ofrece una revisión completa de la red de sensores inalámbricos (WSN) y diversos problemas de calidad de servicio (QoS) en dichas redes. Se centra en cuatro de los principales problemas que están agrupando nodos de sensores desplegados, agregación de datos, paquetes de datos de enrutamiento adecuados y la convergencia de mejora con sensores móviles en un WSN.

### 2.1. Introducción.

WSN ha sido un área de investigación atractiva y se ha utilizado para diversas aplicaciones. WSN integra la comunicación de baja potencia y consiste en un gran número de nodos sensores que se comunican entre sí. Los nodos sensores pueden desplegarse aleatoriamente o manualmente dependiendo de las aplicaciones. Además de los nodos sensores, las WSNs necesitan una o más estaciones base (*Base Station, BS*) para poder realizar comunicaciones y recopilar datos de los sensores desplegados en las áreas monitorizadas.

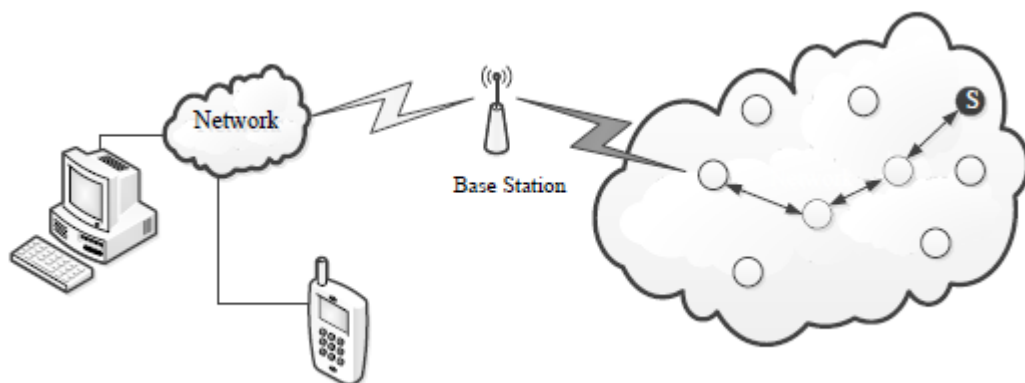


Figura 2. 1: Vista general de una red de sensores inalámbricos.

Fuente: (Al-Karaki & Kamal, 2004)

El papel de la BS es mantener la comunicación entre la red de sensores y la fuente externa (usuarios). De hecho, las comunicaciones entre los nodos sensores y la BS proporcionan a los usuarios la accesibilidad de la información



desde cualquier ubicación remota para permitir la recogida y el análisis de los datos del sensor. En la figura 2.1 se muestra una vista general de una WSN.

Un nodo de sensor inalámbrico es un transductor muy pequeño que es capaz de convertir fenómenos físicos tales como sonido, luz y temperatura en señales eléctricas. Los componentes técnicos se componen de interfaces de sensores, circuito, microcontrolador, batería y sistema de radio. Cada componente y sistema en el dispositivo eléctrico debe funcionar correctamente para lograr el resultado esperado del dispositivo.

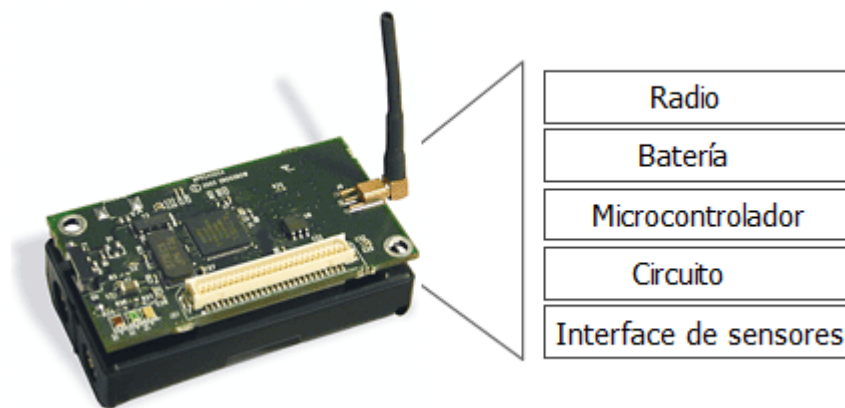


Figura 2. 2: Elementos básicos de un nodo sensor.  
Fuente: (Molina Arribere, 2017)

## 2.2. Visión general.

El capítulo describe las bases fundamentales de las redes de sensores inalámbricos. También se revisa el módulo ZigBee para WSN, y también la radio de comunicación XBee, donde se muestra la arquitectura de la radio de comunicación, el formato de los mensajes enviados y recibidos, la configuración y los modos de operación de la radio. Posteriormente, se presenta el microcontrolador ATmega 328 para controlar la WSN que también trabaja con la radio XBee, el funcionamiento de las puertas de entrada y salida y la administración de energía del microcontrolador, siendo descritas también las placas Arduino y el Shield XBee, aunque éstas no se utilizan en el desarrollo del trabajo de titulación.

Shue, Johnson, & Conrad, (2017) indican que la localización es un atributo importante para las redes de sensores inalámbricos. En la que se puede utilizar el indicador de intensidad de señal recibida (*Received Signal Strength Indicator, RSSI*) para estimar la distancia entre transceptores. Utilizando estas distancias estimadas, la localización de los nodos dentro de una red se puede determinar usando varios algoritmos de localización, tales como trilateración.

### **2.3. Red de Sensores Inalámbricos**

Las redes de sensores inalámbricos (WSNs), es una tecnología de comunicación inalámbrica que utiliza dispositivos de red de bajo consumo de energía. Estos dispositivos de red, también denominados sensores, transmiten datos recopilados por sensores, permitiendo sensar el ambiente en cual ese equipo está insertado, con poco o ningún impacto físico en ese ambiente. Las WSNs difieren de las redes tradicionales Ad-hoc e inalámbricas (WiFi y WiMax) por diversos factores: limitación de energía, ubicación dinámica, no poseen una topología previamente definida, pueden ser extremadamente densas y trabajan de una forma en la que un nodo sensor reenvía la información a otro nodo sensor hasta que esa información llegue al destino.

Un nodo sensor está compuesto por una radio de comunicación para transmisión y recepción de datos, conectado a una antena, y un microcontrolador para controlar la entrada y salida de datos, que efectúa el procesamiento de las señales recibidas de los sensores a través de conversores analógicos-digitales. Un módulo de nodo sensor puede también poseer dispositivos actuadores, como por ejemplo relés, lámparas LED o motores, pudiendo activarlos y desactivarlos de acuerdo con los datos recibidos por la radio. Por último, la batería, que proporciona la energía necesaria para el funcionamiento de todo el módulo, tal como se muestra en la figura 2.3.

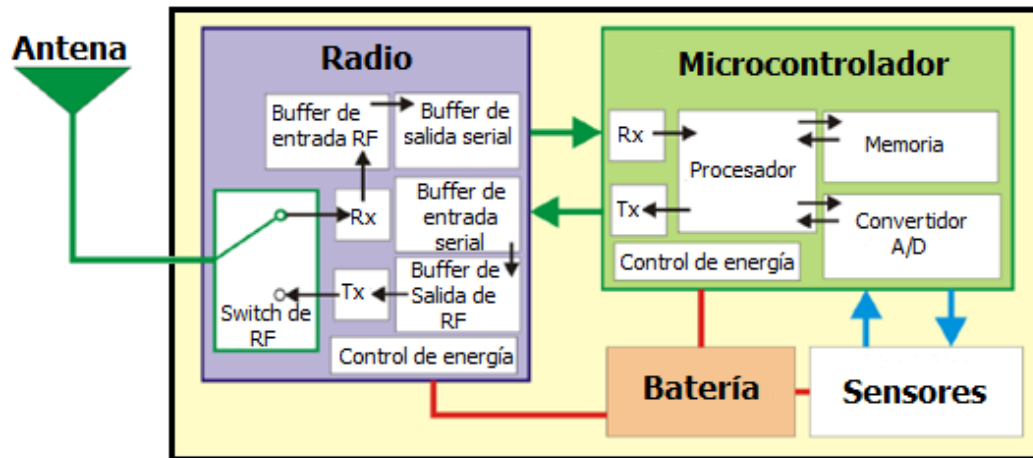


Figura 2. 3: Componentes de un módulo de nodo Sensor.  
Fuente: (Carrión M., Maya O., Domínguez L., & Peluffo O., 2016).

Cuando un dato es recogido por el sensor, pasa por el convertidor analógico digital y está preparado para ser enviado por radio (Tx= transmisor). Los radios de comunicación tienen buffers tanto para envío como para recepción de datos, permitiendo controlar el flujo de transmisión y recepción (Rx=receptor). La radio convierte las señales recibidas por la antena de radio frecuencia (*Radio Frequency, RF*) en bytes para transmitirse de forma serial.

Tanto la radio como el microcontrolador poseen sistemas para el control y administración de energía, permitiendo el ahorro de energía cuando no están en uso. La recepción y la transmisión de RF por la antena no ocurren de forma simultánea, por lo que se utiliza un conmutador (Switch) de RF para cambiar la radio a los modos de recepción o de transmisión. El componente que consume más energía en una red de sensores inalámbricos, sería en si la radio de comunicación. En la que transmitir un bit de información, en términos de consumo de energía, es equivalente al procesamiento de 3000 instrucciones.

En redes de sensores inalámbricos, una solución para ahorrar energía es analizar y procesar la información de los sensores en el microcontrolador del módulo, transmitiendo sólo los datos que se ajustan a los requisitos del

proyecto de la red, teniendo también un importante papel en la agregación de datos. En este trabajo, el microcontrolador realiza el tratamiento de los datos antes de la transmisión, permitiendo ahorrar energía al enviar sólo informaciones que sean relevantes en el sistema de gestión de la red.

### **2.3.1. Aplicaciones redes de sensores inalámbricos.**

Las aplicaciones de WSN en el dominio de detección de eventos se pueden clasificar en cinco categorías principales. En esta sección algunas de las aplicaciones van a ser exploradas. En primer lugar, las aplicaciones de entorno en las que se despliega un gran número de nodos de sensores en un área de interés. Las WSNs son responsables de:

- a) monitorización de la calidad del agua mediante STORM 3 Data Loggers [investigación publicada por los autores Sun, Ahmed, Sun, Qian, & Xiao, (2016)]
- b) medición de la temperatura de fibra óptica distribuida en áreas de minas de carbón [investigación publicada por los autores Zhao, Li, Zhang, & Zheng, (2010)]
- c) localización y detección de emisiones acústicas del oleoducto [investigación publicada por los autores Wang, Jiao, Yang, & Niu, (2016)]
- d) monitorización de pulso y temperatura de pacientes basados en ZigBee [investigación publicada por Agajo, (2016)]
- e) monitorización del tránsito vehicular y de contaminación CO<sub>2</sub> [trabajo realizado por los autores Granda Bravo & Belduma Belduma, (2017)]
- f) monitoreo en aplicaciones domóticas [trabajo realizado por los autores Cedeño Villarroel & Eras Pérez, (2010)]
- g) control de los signos vitales en pacientes adultos mayores [trabajo realizado por Ruiz Sánchez, (2016)]
- h) monitoreo continuo de la humedad del suelo [investigación desarrollada por los autores Flores-Medina, Flores-García, Velasco-Martínez, González-Cervantes, & Jurado-Zamarripa, (2015)]

Las WSNs también pueden aplicarse en campos industriales con fines de monitoreo. La principal motivación de usar WSNs en lugar de las redes de sensores con cable en la industria es la flexibilidad y la capacidad de auto organización. En el caso de agregar o quitar un nodo de sensor de una WSN, las redes pueden reconfigurarse sin preocuparse por el cableado. Además, los nodos sensores pueden colocarse en una parte móvil de maquinarias y áreas inaccesibles para monitoreo remoto, donde los sensores cableados pueden no ser capaces de aplicar.

Las WSNs se utilizan generalmente para detectar el rendimiento y las fallas de funcionamiento y para detectar problemas de seguridad en las grandes plantas industriales, la automatización de edificios y la integridad estructural de vigilancia. Como resultado de la monitorización y detección de fallos o anomalías en la industria se podrían evitar posibles daños en maquinarias que pudieran ser muy costosas. Además, desde el punto de vista de la seguridad personal, las WSNs pueden ayudar a los operadores que trabajan en la industria.

### **2.3.2. Detección de eventos en redes de sensores inalámbricos.**

La detección de eventos mediante la tecnología WSN ha sido un área de investigación en las últimas décadas. Esto se debe a las diversas aplicaciones en el mundo real. Los eventos en WSNs corresponde a un fenómeno del mundo real que ocurre en los entornos que se están supervisando. El proceso de detección de eventos generalmente se puede categorizar en dos principales clases de detección, que son: centralizada y distribuida.

En sistemas basados en la detección centralizada, cada sensor debe enviar sus observaciones a un centro o a una estación base sin procesar o perder información. Entonces, el centro estaría a cargo de procesar los datos recibidos y darse cuenta de que si incluso ocurrió. Obviamente, si aumenta el tráfico, es lógico que también el consumo de energía, ya que la transferencia de datos consume más energía que su procesamiento. Eso es claramente

peor en redes de sensores de gran escala en aplicaciones del mundo real con la gran cantidad de transmisiones de fuentes a la estación base.

Esto también supone un retardo significativo en el proceso de detección de eventos en las WSNs. Por lo tanto, el esquema de detección de eventos descentralizados o distribuidos puede funcionar mejor ya que los nodos sensores están limitados por la limitada potencia y la comunicación de ancho de banda. En la arquitectura de detección descentralizada, los nodos sensores en vez de enviar paquetes de datos directamente a la estación base, deciden la ocurrencia de un evento, en base a los datos detectados.

Por ejemplo el trabajo de (Huang, Sharma, & Shah, 2011) presenta una ventana dinámica optimizada basada en investigaciones anteriores y los resultados de la simulación mostraron que el tiempo medio de cálculo, debido a las condiciones difusas, disminuyó aproximadamente del 18% en relación a otros algoritmos en una red de sensores inalámbricos de criptosistemas de curva elíptica.

Mientras que Kapitanova, Son, & Kang, (2012) utilizan lógica difusa para la detección robusta de eventos en redes de sensores inalámbricos. Es decir, que mediante el uso de valores difusos en sitios frescos mejoraron significativamente la precisión de la detección de eventos.

#### **2.4. Calidad de servicio en redes de sensores inalámbricos.**

Para Mbowe & Oreku, (2014) la calidad de servicio (*Quality Service, QoS*) es un nivel medible de servicio entregado a los usuarios de la red, que puede caracterizarse por la probabilidad de pérdida de paquetes, ancho de banda disponible, retardo de extremo a extremo (end-to-end), etc. La QoS puede ser proporcionada por los proveedores de servicios de red en términos de algún acuerdo de nivel de servicio (*Service Level Agreement, SLA*) entre los usuarios de la red y los proveedores de servicios. Por ejemplo, los usuarios

pueden requerir que, para algunos flujos de tráfico, la red elija una ruta con un ancho de banda mínimo.

Diferentes técnicas en WSNs pueden reconocer QoS de diferentes maneras, que se pueden diseñar en función de los requisitos de las aplicaciones. Por ejemplo, en un sistema de control seguro, el retardo y la pérdida de paquetes pueden no ser permitidos, mientras que podría ser aceptable en sistemas de aire acondicionado en una oficina. Sin embargo, los métodos QoS en cualquier aplicación se aplican para mejorar el rendimiento de WSNs

Mbowe & Oreku, (2014) en su artículo presentan las métricas apropiadas de QoS para WSN que implican servicio, fiabilidad y disponibilidad que en última instancia facilitan el archivado del servicio cualitativo. En resumen, ellos discutían el aspecto inverso de QoS y por lo tanto presentaron matemáticamente tres factores de calidad significativos que deben ser tomados en cuenta para el desarrollo de servicios de calidad de aplicaciones WSN, es decir, disponibilidad, fiabilidad y facilidad de servicio.

## **2.5. Estándares de ZigBee**

Esta sección presenta el estándar ZigBee, estándar que se desarrolla sobre la base del protocolo IEEE 802.15.4. El funcionamiento de ZigBee se describe en la sección 2.3.2, donde se presentan las topologías utilizadas y la descripción de los componentes de la red. En la sección 2.3.3, se muestra la arquitectura en capas del patrón ZigBee.

### **2.5.1. Protocolo IEEE 802.15.4.**

El protocolo IEEE 802.15.4 fue ratificado por el Instituto de Ingenieros Eléctricos y Electrónicos (*Institute of Electrical and Electronics Engineers, IEEE*) como estándar para redes de baja velocidad y bajo consumo de energía, siendo por esta razón muy utilizado en WSNs. Las redes IEEE 802.15.4 operan en el rango de frecuencia de 2,4 GHz (es decir, en el rango

comprendido entre 2400 a 2483,5 MHz), y con una velocidad de transmisión de 250 kbps.

Otras bandas de frecuencia son permitidas por el protocolo, con variación del número de canales de comunicación y de la velocidad de transmisión, tal como se puede observar en la tabla 2.1. El protocolo IEEE 802.15.4 implementa capas tanto, física (PHY) y de control de acceso al medio (*Medium Access Control, MAC*), dejando las capas de red y superiores libres para su implementación de acuerdo con el uso. La capa física es la responsable de la interconexión del protocolo con la radio de comunicación y provee servicios para la capa inmediatamente superior, que controla el acceso a la red, la capa MAC.

Tabla 2. 1: Rangos de frecuencia del protocolo IEEE 802.15.4.

Frecuencia	868,3 MHz	902-928 MHz	2400 a 2483,5 MHz
N° de Canales	1	10	16
Velocidad de datos (kbps)	20	40	250
Ubicación	Europa	América	Mundial

Fuente: (IEEE, 2015)

Como el estándar IEEE 802.15.4 define sólo estas dos capas, permite que otros protocolos lo utilicen como base para la implementación de capas superiores, como es el caso del protocolo ZigBee. El protocolo IEEE 802.15.4 opera con dos tipos de dispositivos físicos: el dispositivo de función completa (*Full Function Device, FFD*), que es responsable del enrutamiento en la red, siendo que un nodo FFD se utiliza como coordinador de la red, controlando la topología y direccionamiento de los nodos y como enrutador, haciendo el reenvío de mensajes; y el dispositivo de función reducida (*Reduced Function Device, RFD*), que es un dispositivo de bajo costo, no hace enrutamiento y sólo puede conectarse a un nodo FFD a la vez.

El encaminamiento permite que un nodo reenvíe un mensaje de un nodo RFD o FFD a otros nodos de FFD hasta el nodo coordinador y normalmente



tanto los nodos FFD para enrutamiento como los nodos FFD coordinadores necesitan una fuente de energía constante, ya que necesitan estar activos en todo momento para la transmisión de mensajes, a diferencia de los nodos RFD que normalmente son alimentados por baterías y necesitan ahorrar la mayor cantidad de energía posible.

### 2.5.2. Funcionamiento del ZigBee

De acuerdo al trabajo de Gifty & Sumathi, (2016) la red ZigBee puede configurarse como tres topologías principales, como la topología en estrella, árbol o malla. El objetivo principal es tener bajo consumo de energía y alta utilización de canales, el impacto de los parámetros de MAC para la topología de estrella, árbol y malla en el modo de desactivación y modo de habilitación.

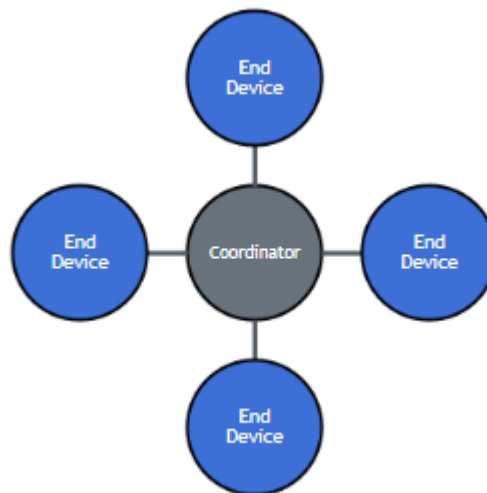


Figura 2. 4: Topología estrella de una red ZigBee.  
Fuente: (Cardell, 2017; Hillman, 2017)

El estándar ZigBee fue especificado por ZigBee Alliance en 2006 y utiliza las capas física y MAC del protocolo IEEE 802.15.4, añadiendo las capas de red y aplicación, además de encriptación AES de 128 bits y puede dirigirse hasta 65535 nodos por subred (IEEE, 2015). En la figura 2.4 se muestra el tipo de topología estrella, donde el nodo coordinador (*Coordinator, C*) ZigBee y el enrutador (*Router, R*) ZigBee se implementan como FFD y el dispositivo final (*End Device, E*) ZigBee se implementan como nodo RFD, ya que no

realiza enrutamiento. En general, los nodos de dispositivo final ZigBee se denominan nodos sensores.

Y de manera similar, en las figuras 2.5 y 2.6 se muestran las topologías en árbol (Tree) y enmallada (Mesh). De acuerdo a Hillman, (2017), la figura 2.5 muestra la topología de árbol, donde C forma el nodo raíz de un árbol de nodos secundarios. Los dispositivos finales (E) son nodos rama (aunque un enrutador también podría ser una rama si aún no se han unido todavía) y los nodos intermedios son enrutadores (R). La comunicación directa sólo puede ocurrir entre un nodo esclavo y su maestro, pero todos los nodos pueden comunicarse entre sí mediante mensajes que atraviesan el árbol hasta un antepasado común y luego al nodo objetivo. En esta topología, los enrutadores pueden ampliar el rango de la red más allá de cualquier enlace de dispositivo a dispositivo. Sin embargo, si falla un enrutador, no hay ruta alternativa y partes de la red pueden desconectarse.

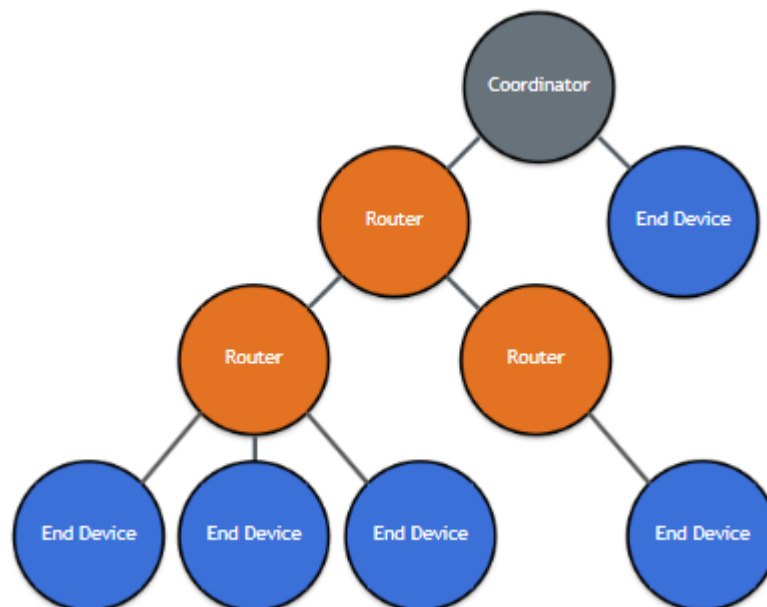


Figura 2. 5: Topología estrella de una red ZigBee.  
Fuente: (Cardell, 2017; Hillman, 2017)

La topología de malla (véase la figura 2.6) es una de las más flexibles que ofrece ZigBee. Es similar a la topología de árbol, pero sin seguir la estructura de árbol rígida y un enrutador puede comunicarse directamente con

cualquier otro enrutador o el coordinador si está en el rango. Esto significa que puede haber muchas rutas diferentes a través de la red a un nodo dado, y ZigBee tiene una característica de descubrimiento de ruta para encontrar la mejor ruta a un nodo dado. (Hillman, 2017)

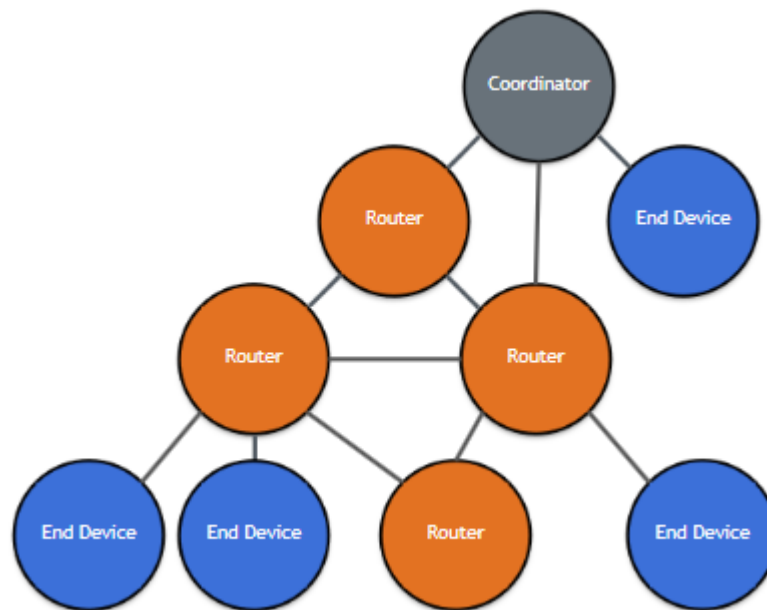


Figura 2. 6: Topología enmallada de una red ZigBee.  
Fuente: (Cardell, 2017; Hillman, 2017)

De manera general, el funcionamiento de una red ZigBee se describe a continuación: (a) el nodo coordinador inicializa la red seleccionando un canal de comunicación y un identificador para esa red, denominado PAN ID; (b) el nodo coordinador permite que los enrutadores y los sensores ingresen en la red, ayudando al enrutamiento de datos y el almacenamiento de paquetes a los nodos que no están activos.

Por lo tanto, el nodo coordinador debe estar siempre activo y con una fuente constante de alimentación. Los nodos enrutadores pueden comenzar a transmitir, recibir o enrutar datos después de haber ingresado en una red iniciada por el coordinador. Los nodos de enrutadores también permiten que otros enrutadores y los sensores ingresen a la red. Al igual que el nodo coordinador, los nodos de enrutadores deben estar siempre activos debido al

enrutamiento y al almacenamiento de paquetes destinados a los nodos no activos. (RF Wireless World, 2017)

Los nodos sensores deben ingresar en una red a través de un enrutador o de un coordinador antes de comenzar a transmitir y recibir datos. Otros nodos que no pueden ingresar en la red a través de un nodo del sensor y, a diferencia de los nodos de coordinador y enrutador, el nodo del sensor siempre debe transmitir y recibir datos a través del nodo por el que ingresó a la red (nodo padre). El nodo sensor también puede entrar en modo de ahorro de energía para aumentar la vida de la red.

En la red ZigBee, sólo existe un nodo coordinador por red, independientemente del tipo de topología utilizada. Al iniciar una red, el nodo coordinador establece un identificador único de la red (PAN ID). Este identificador será común a todos los demás nodos que ingresan en esa red, tal como se muestra en la figura 2.7 una aplicación general de la topología ZigBee.

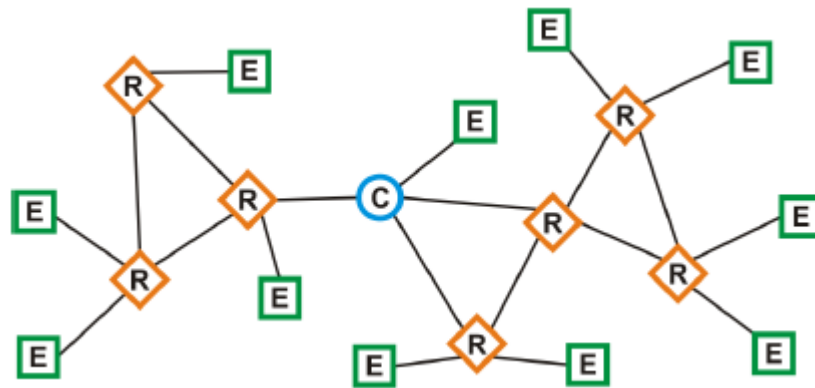


Figura 2. 7: Ejemplo de aplicación para la topología de una red ZigBee.  
Fuente: (RF Wireless World, 2017)

Normalmente, los dispositivos ZigBee se preconfiguran con una dirección PAN ID para agregarlos a una red existente, pero también pueden seleccionar un PAN ID de alguna red cercana para solicitar el ingreso. El PAN ID se compone de dos direcciones, uno de 16 bits y otro de 64 bits. La cantidad de combinaciones de direcciones de 16 bits (65535 posibilidades) hace que

exista la posibilidad de que dos redes diferentes utilicen el mismo ID y por este motivo también se utiliza la dirección de 64 bits como identificador de red.

### **2.5.3. Capas de ZigBee.**

Esta sección presenta la arquitectura de capas de la red ZigBee, importante para entender el funcionamiento del protocolo y su implementación sobre el protocolo IEEE 802.15.4. En el trabajo de Alaparthy & Rao, (2016) describen el funcionamiento de las capas física y MAC del protocolo IEEE 802.15.4, además de citar como la capa de red se implementa junto con el protocolo ZigBee.

Tanto el protocolo IEEE 802.15.4 como el protocolo ZigBee fueron objeto de un análisis y de una simulación en escenarios a gran escala en el trabajo realizado por Biddut, Islam, Sultana, Sarker, & Rahman, (2015). En la figura 2.8 se muestra la estructura de capas de la red ZigBee, donde se puede ver la implementación del estándar ZigBee sobre el protocolo IEEE 802.15.4, tal como se pueden revisar los trabajos de (Alaparthy & Rao, 2016) y (Biddut et al., 2015).

En la capa física se define la sensibilidad de recepción, el número de canales, el rechazo de un canal y la especificación de la velocidad de transmisión de datos. Las redes ZigBee operan en el rango de frecuencia de 2, 4 GHz y con una velocidad de transmisión de 250 kbps, igual que el protocolo IEEE 802.15.4 (véase la tabla 2.1). En la capa MAC se realiza la gestión de la transmisión de datos entre nodos vecinos, por difusión, incluyendo el servicio de reenvío de transmisiones y las técnicas de prevención de colisiones CSMA/CA. Aunque, también incluye los servicios de asociación y desasociación de nodos en la red.

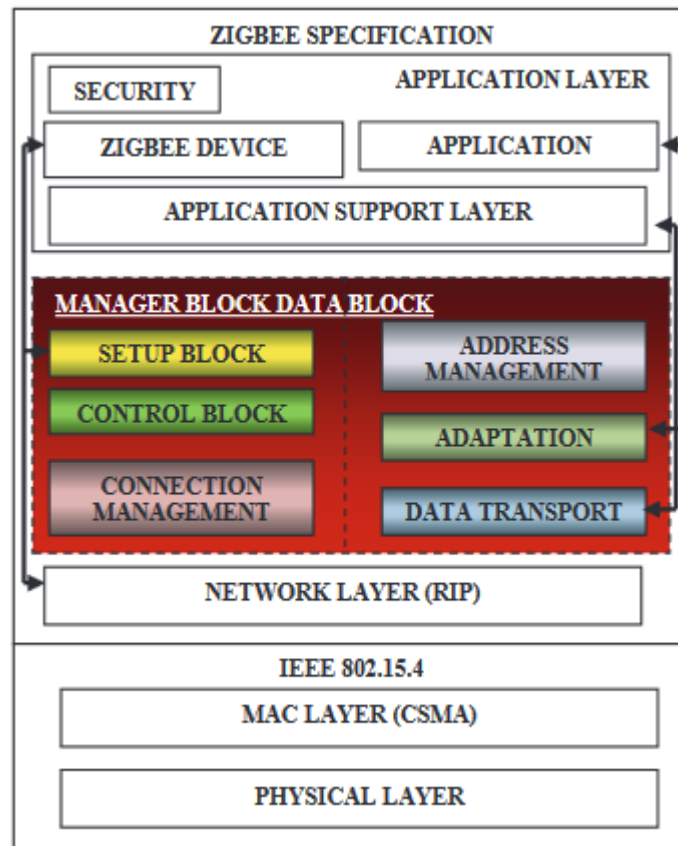


Figura 2. 8: Capas ZigBee e IEEE 802.15.4.  
Fuente: (Alaparathi & Rao, 2016)

La red ZigBee puede trabajar con varios tipos de enrutamiento, por ejemplo en el trabajo realizado por Gifty & Sumathi, (2016) realiza el análisis comparativo de modelos de energía para WSN utilizando ZigBee en el consumo de energía en modo de transmisión, recepción y modo inactivo. En su trabajo presentan una breve descripción de las topologías estrella (star), árbol (tree) y malla (mesh), así como los protocolos de enrutamiento AODV y DSR. En la cual, cada topología se realizan configuraciones del rendimiento de los esquemas de enrutamiento de protocolos sobre la base de métricas de los parámetros de rendimiento, tales como: fluctuación (jitter) promedio, retardo (delay) promedio y rendimiento.

(Khalaf & Mokadem, 2016) en su trabajo analizaron dos escenarios comparativos que incluye mediciones de tráfico de datos recibido, enviado, rendimiento y retardo. Las simulaciones fueron realizadas en Riverbed

Modeler Academic Edition 17.5: (a) 1er escenario, las tres topologías (estrella, árbol y malla) se comparan entre sí en el caso de varios fallos de nodos como coordinador, enrutador y dispositivo final para certificar la fiabilidad de estas redes. Sólo hay un coordinador ZigBee en cada topología. La comparación incluye las estadísticas de tráfico de datos enviado, tráfico de datos recibido, rendimiento y retraso. (b) 2do escenario, se elige la topología de árbol con un solo nodo C y se compara con una red similar que tiene un nodo C adicional. La red de dos nodos C forma dos PANs.

## **2.6. Radio XBee**

Esta sección presenta la radio XBee, que, aunque no fue utilizada en este trabajo como radio de comunicación de la WSN. Los módulos XBee son descritos brevemente para fundamentar la teoría relacionada a las redes de sensores inalámbricos que utilizan comunicación XBee. El módulo de radio XBee en sí es un transceptor inalámbrico. XBee dispone de varios módulos, para implementaciones tales como, WiFi, ZigBee, y ZigBee 800 MHz.

XBee se convertiría en un estándar con los fabricantes, para lo cual XBee era barato y lo suficientemente versátil para satisfacer las especificaciones deseadas. Los módulos de la serie 2 de ZigBee fueron elegidos por sus requerimientos de energía de bajo costo y mínimos. También funcionan en el rango de frecuencias estándar de 2,4 GHz, lo que significa que el hardware podría eventualmente construir fuera de las infraestructuras existentes como el XBee ZigBee que utiliza el mismo protocolo, es decir, IEEE 802.15.4, y de igual manera sucede con los enrutadores inalámbricos.

(Shue et al., 2017) en su trabajo presenta una plataforma basada en los módulos inalámbricos XBee ZigBee, Arduino y MatLab en el que realizan pruebas de algoritmos. Esta evaluación fue validada a través del método de localización de trilateración. En la cual se puede demostrar el empleo del módulo XBee en aplicaciones WSNs.

Mientras que los autores Iswandi, Nastiti, Praditya, & Mustika, (2016) en su trabajo ofrecen un estudio empírico del rango de transmisión XBee en el bosque basado en RSSI. Estas mediciones fueron realizadas en dos tipos diferentes de bosque, es decir, bosque tropical heterogéneo y bosque de pino homogéneo. En la figura 2.9 se muestra el esquemático de conexiones de los equipos que se utilizan en transmisión y recepción. Este trabajo pudo demostrar a través de mediciones, una alta degradación del rango de transmisión en comparación con el ambiente no forestal. Es importante señalar que la distancia máxima de transmisión fue entre 250 y 330 m para ambos escenarios.

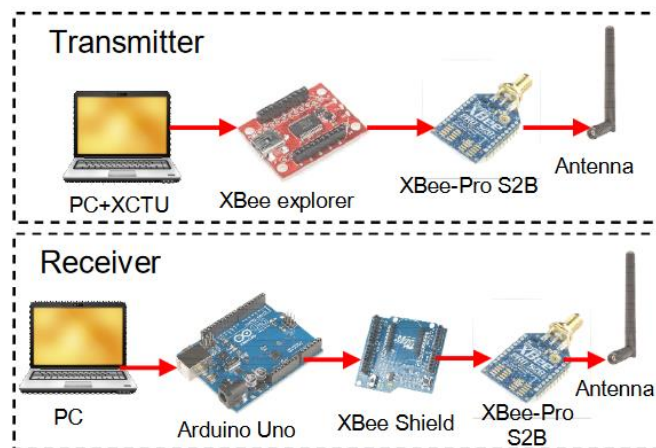


Figura 2. 9: Esquemático de la configuración de equipos utilizados para pruebas.  
Fuente: (Iswandi et al., 2016)

En la figura 2.10 se muestran los módulos o tipos de antenas de XBee, que son: antena de cable (*Whip Antenna*), conector estándar u.FL (*u.FL. RF Connector*) para antena externa (conector coaxial en miniatura), antena de chip (*Chip Antenna*) y conector RPSMA.

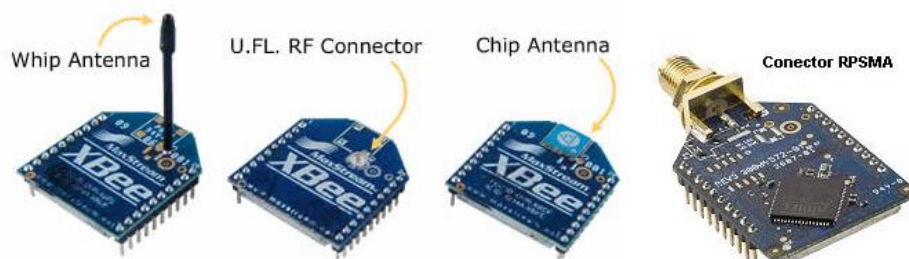


Figura 2. 10: Tipos de antenas para XBee.  
Fuente: (XBee, 2017)



### 2.6.1. Formato de envío y recepción de datos

La radio XBee trabaja con dos modos de operación de transmisión y recepción de datos. En el primer modo, comando de atención (AT), los datos son enviados y recibidos directamente por el puerto serial, teniendo una interfaz sencilla y siendo más fácil el desarrollo de aplicaciones, bastando la aplicación conectarse al puerto serial del módulo y enviar los datos utilizando comandos AT. Este modo, aunque simple, no es escalable para enviar datos a múltiples destinatarios y tampoco permite el envío de configuraciones remotas de módulos.

El segundo modo, es el modo de interfaz de programación de aplicaciones (*Application Programming Interface, API*) que basa en el envío y recepción de tramas (frames) de datos, especificación como comandos, respuestas de comandos y mensajes sobre el estado de funcionamiento del módulo (enviados y recibidos), permitiendo un aumento en la escalabilidad de la red.

Para Ortega-Corral, Palafox, García-Macías, Sánchez-García, & Aguilar, (2014) los comandos AT también se pueden enviar y recibir a través del modo API, permitiendo la coexistencia de los dos modos en una red. La estructura de la tabla de datos se describe en la figura 2.11 y las aplicaciones que utilizan el modo API deben ajustarse a esta estructura.

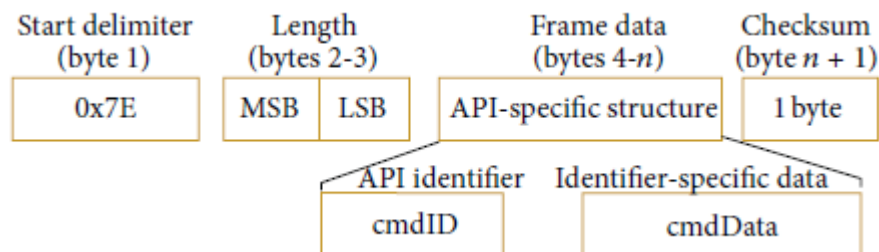


Figura 2. 11: Tabla de datos del modelo de tramas API.

Fuente: (Ortega-Corral et al., 2014)

La tabla de datos API empieza siendo un delimitador inicial (Start delimiter) 1 byte con valor 0x7E; seguido de dos bytes con el tamaño de trama

de datos, dividido entre el bit más significativo (*Most Significant Byte, MSB*) y el bit menos significativo (*Least Significant Byte, LSB*). Mientras, que para el tamaño de la trama de datos no está incluido el último byte, de checksum. Dependiendo del tipo de API, el tamaño de la trama de datos puede variar significativamente.

En la figura 2.12, la trama API específica corresponde a una respuesta automática de XBee E/S remota identificada por el identificador de API 0x83 (cmdID), que en nuestro caso es una trama de señal de entrada direccionada XBee de 16 bits enviada automáticamente por un EP y recibida por un Nodo CLH de destino. Una carga útil de la trama 0x83 puede contener muestras A/D procedentes de hasta seis entradas analógicas de la fuente remota EP, entre otras características de detección de entrada.

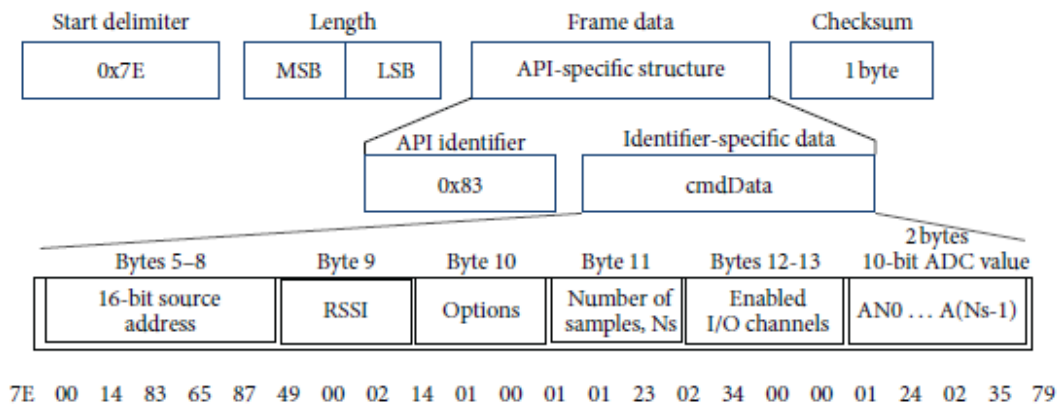


Figura 2. 12: Modelo la API de identificación 0x83.

Fuente: (Ortega-Corral et al., 2014)

### 2.6.2. Configuración del radio XBee

La configuración del módulo XBee se realiza a través del programa XCTU, desarrollado por el fabricante del radio XBee. En la figura 2.11 se muestra como realizar configuración del módulo. Es necesario inicialmente seleccionar el puerto serial, la velocidad de transferencia (baudios/bps), control de flujo (*flow control*), los bits de datos (*data bits*), paridad (*parity*) y los bits de parada (*stop bits*).

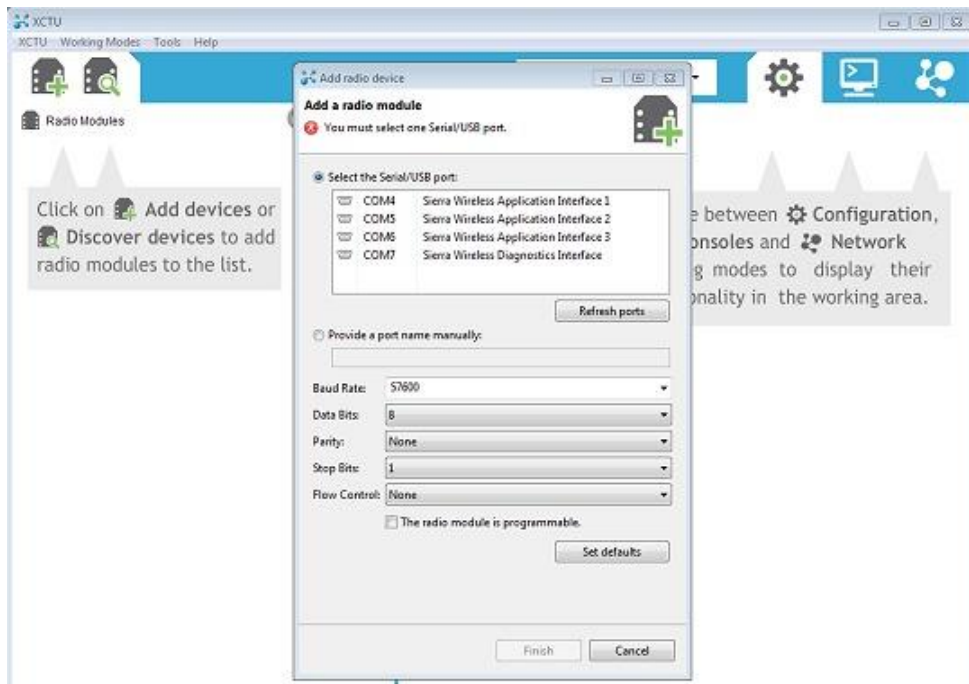


Figura 2. 13: Configuración del módulo de una radio.  
Fuente: (Hobbytronics, 2017)

### 2.6.3. Puertos de Entrada/Salida.

El módulo XBee tiene 13 puertos de entrada/salida (DIO0 a DIO12), algunos poseen más de una función, que son configurados a través del programa XCTU. En la figura 2.14 se muestra el diagrama de pines del módulo XBee y sus respectivas funciones. Se puede ver que para la alimentación del módulo se usa el pin 1 (VCC) con un valor mínimo de 3,3 V y la tierra (GND) corresponde al pin 10.



Figura 2. 14: Pinza del radio XBee.  
Fuente: (Farnell, 2007)

La entrada y salida de datos se realizan a través de las puertas DOUT (pin 2) y DIN (pin 3), respectivamente. Estos pines se pueden conectar al microcontrolador para enviar y recibir datos de los sensores. La función RESET del módulo XBee se ejecuta conectando un pulsador entre el pin 5 (RESET) y el pin 10 (Ground). El pin 6, además de la función DIO10, puede ser utilizado para medir la señal RSSI de la última transmisión recibida por el módulo.

El pin 8 es de uso reservado, no debiendo ser conectado. El pin 9, además del puerto DIO8, se puede utilizar para activar/desactivar el estado de inactividad del módulo. Esta puerta, cuando existe un nivel de tensión aplicado a ella coloca el módulo en el estado de inactividad y, cuando está a nivel bajo o conectado al pin 10 (Ground), el módulo pasa a operar normalmente. Los pines 4 (DIO8), 7 (DIO11) y 11 (DIO4) se utilizan como entradas o salidas digitales y sólo tienen esta función. Los pines 12 (CTS/DIO7) y 16 (RTS/DIO6) se pueden utilizar para el control de flujo listo para enviar (*Clear To Send, CTS*) y solicitud de envío (*Request to Send, RTS*), respectivamente.

El pin 13 (ON/SLEEP), además de la función de E/S Digital 9 (DIO9), puede utilizarse como indicativo del estado de funcionamiento del módulo, utilizando un LED que, cuando está encendido, indica el funcionamiento normal y cuando está apagado indica que el módulo se encuentra en estado de reposo (SLEEP). El pin 15 tiene un funcionamiento similar, pero indicativo del estado de la asociación del módulo con la red, además de la función E/S Digital 5 (DIO5). Si el módulo no está asociado a una red, el LED permanece encendido. Cuando el módulo está asociado a una red, debe parpadear en un intervalo regular, de 500 ms para nodos coordinadores y de 250 ms para nodos de enrutadores y dispositivos finales.

El módulo XBee también tiene cuatro pines que pueden utilizarse como E/S digital o solo como entrada analógica, de los pines 17 a 20,

respectivamente, AD3/DIO3, AD2/DIO2, AD1/DIO1 y AD0/DIO0. El último pin 20, también se puede utilizar como pulsador de puesta en marcha cuando se conecta un pulsador y éste al pin 10 (Ground). Por ejemplo, cuando el módulo está en una red, al presionar el botón una vez, un módulo dispositivo final se despierta durante 30 segundos y envía una transmisión de difusión con la identificación del nodo.

Al presionar el botón dos veces, el módulo envía una transmisión de difusión permitiendo al nodo coordinador y a todos los demás nodos ingresar a la red durante 1 minuto. Presionando cuatro veces el pulsador, el módulo deja la PAN, restaurando parámetros de ID de la red y de búsqueda de canales (*Search Channel, SC*) a los valores por defecto e intenta conectarse a PAN utilizando esos valores.

#### **2.6.4. Modos de funcionamiento**

En la figura 2.15 se muestran los cinco modos de funcionamiento de la radio XBee: inactivo (Idle), transmisión (Transmit), recibir (Receive), comando (Command) y sueño (Sleep). Se mantiene de forma predeterminada en el modo Idle cuando no está en operación de envío y recepción de datos. A partir de ese modo, puede cambiar a los modos de transmisión cuando necesita transmitir datos; recibir, cuando un paquete de datos válido es recibido por la antena; comando, cuando está en modo para recibir instrucciones de control y configuración de la radio y sueño, en que la radio se queda en estado de dormición.

La radio XBee tiene el parámetro de configuración SM, que especifica el modo de funcionamiento de modo de suspensión. A continuación, se explican los modos de ahorro de energía. Los parámetros se cambian utilizando el software X-CTU.

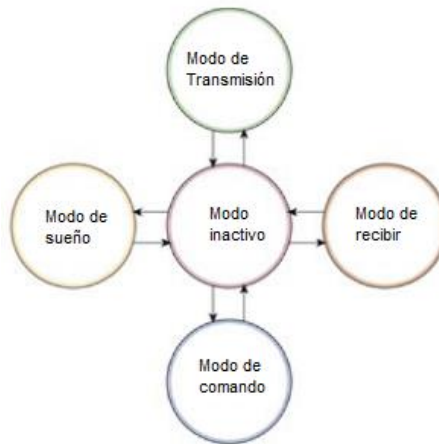


Figura 2. 15: Modos de funcionamiento del XBee.  
Fuente: (Digi, 2017)

## 2.7. Plataforma Arduino.

La plataforma de prototipado Arduino está basada en hardware abierto, es decir, que es posible desarrollar productos utilizando como base el no pago de derechos, permitiendo así la personalización y adaptación de la red de sensores. Todo dependerá de las necesidades de la investigación, permitiendo así un prototipado rápido. El módulo Arduino está compuesto por un microcontrolador ATmega 328, un chip de comunicación USB FTDI para interconexión con un microcomputador y reguladores de voltaje que permiten el uso de fuentes de energía entre 5 y 12 V (véase la figura 2.16).

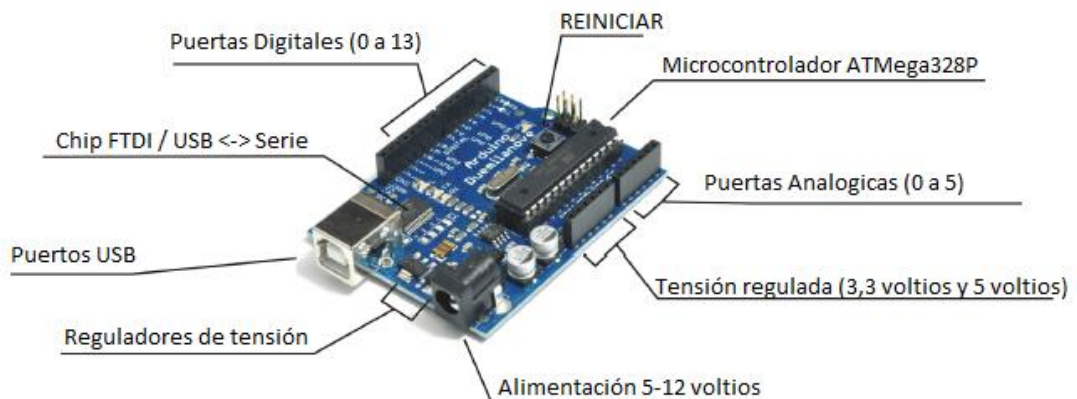


Figura 2. 16: Placa Arduino Duemilanove.  
Fuente: (Cardell, 2017)g

Arduino tiene 14 puertos digitales programables (0 a 13). Los puertos 0 y 1 se utilizan para comunicación serial (ejemplo: como radio de comunicación). Los puertos 2 y 3, además de la función normal de entrada y salida (E/S), son utilizados para interrupciones de un microcontrolador. Si un sensor de presencia está conectado a este puerto, cuando se produce un evento en el sensor, un tratamiento especial puede darse, por ejemplo, enviar un mensaje de emergencia o quitar el microcontrolador del estado de inactividad.

Los otros puertos pueden ser utilizados para sensores que devuelvan valores booleanos, como verdadero o falso, activar y desactivar relés/transistores para controlar dispositivos de mayor potencia o tensión, además de encender y apagar lámparas, diodos emisores de luz (*Light Emission Diode, LED*). Las puertas digitales 3, 5, 6, 9, 10 y 11 se pueden utilizar como puertas de cambio de modulación por ancho de pulso (*Pulse Width Modulation, PWM*).

Además de los puertos digitales, Arduino tiene 6 puertos analógicos, cada puerto con un convertidor A/D (analógico-digital) de 10 bits. Estos convertidores funcionan midiendo el voltaje existente en cada puerto, proporcionando un valor entre 0, sin signo, y 1023, equivalente a 3,3 V o 5 V, de acuerdo con el nivel de voltaje utilizado. Se pueden acoplar a estos puertos sensores como luminosidad, humedad, temperatura, gas y sísmico/acelerómetro. Otros componentes importantes de la placa Arduino son el botón de reset, para el reinicio del módulo, y los conectores con tensión regulada en 3,3 V y 5 V, para uso en sensores y/o placas acopladas a la placa principal del Arduino.

La plataforma ofrece el concepto de shields, que son placas que se pueden añadir a Arduino para aumentar su funcionalidad. Hay blindajes para conexión de Arduino con módulos Bluetooth, Ethernet, relés para control de potencia. El fabricante de Arduino también proporciona un software para el

desarrollo de aplicaciones, basadas en el lenguaje C, permitiendo la programación en alto nivel. En el caso de Arduino, que se utiliza en el puerto serial para la comunicación con el microordenador, hay varias versiones personalizadas de Arduino desarrolladas en diversos países.

## **2.8. Protocolos inalámbricos apropiados para el ambiente y requerimientos de la industria petrolera.**

Esta sección se enfocará en las limitaciones y restricciones más relevantes impuestas por las especificaciones de los protocolos de comunicación inalámbricos industriales estándares diseñados para el monitoreo de redes inalámbricas industriales. Los estándares considerados se listan a continuación:

1. Estándar IEEE 802.15.4: La interface de aire correspondiente a la capa física y a la sub-capa de control de acceso al medio (MAC). Este estándar ya fue descrito en la sección 2.5. dentro del estándar ZigBee.
2. Estándar ZigBee: también fue tratado en la sección 2.5.
3. Estándar WirelessHART.
4. Estándar ISA SP 100.

### **2.8.1. Estándar WirelessHART.**

Kim, Hekland, Petersen, & Doyle, (2008) indican que WirelessHART fue oficialmente publicado en septiembre del 2007, siendo el primer estándar de comunicación inalámbrica abierto para el control de procesos y aplicaciones relacionadas. Podría resumirse como un sistema de red de malla IEEE 802.15.4 basado en el acceso múltiple por división de tiempo (*Time Division Multiple Access, TDMA*) que opera en la banda de radio no licenciada de 2.4 GHz con requerimientos rigurosos de sincronización y seguridad.

Mientras que para Zhang, Yan, & Ma, (2013), WirelessHART es el primer protocolo internacional dirigido a la automatización industrial y al control de procesos. Desde que se anunció en 2007, muchos académicos han llevado a cabo una exploración profunda en la aplicación, las tecnologías clave y el



diseño de la pila de protocolos de WirelessHART. La industria de automatización de procesos ahora tiene acceso a un estándar abierto que ofrece una interfaz inalámbrica a dispositivos de campo, denominada WirelessHART.

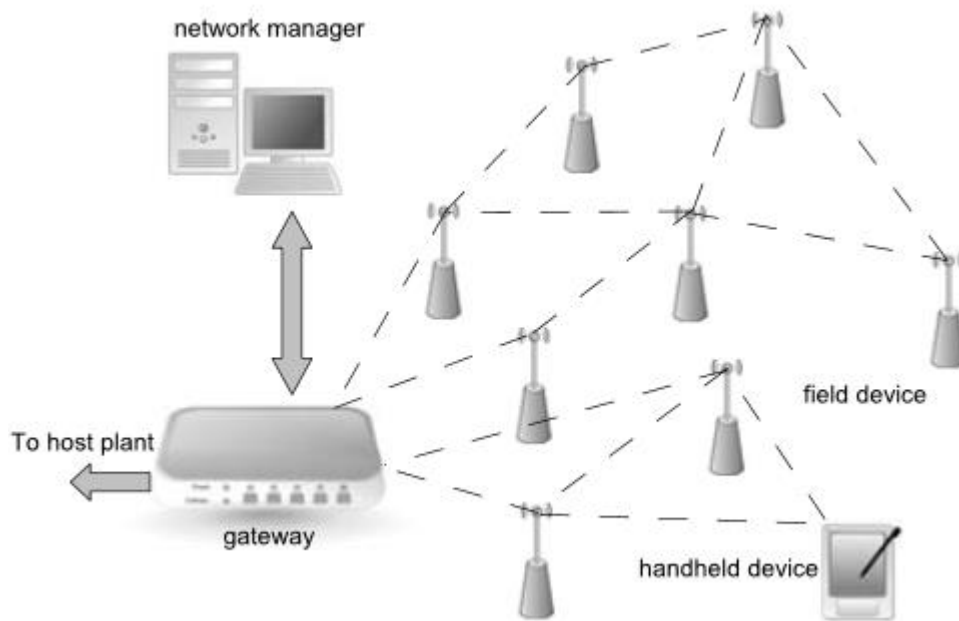


Figura 2. 17: Diagrama de una red WirelessHART genérica.  
Fuente: (Kim et al., 2008)

La figura 2.17 muestra una red WirelessHART, la misma que forma un grupo de dispositivos de red. Pueden ser un dispositivo de campo que está directamente conectado a la planta de proceso, o dispositivos portátiles, por ejemplo, un trabajador de mantenimiento. Se observa en la figura 2.17, que la red admite la conexión directa entre el dispositivo y la pasarela (topología en estrella) y las conexiones a través de múltiples saltos (topología de malla). Por lo tanto, cada dispositivo de red debe poder funcionar como fuente, receptor y enrutador.

WirelessHART fue diseñado especialmente para soportar la amplia gama de aplicaciones de proceso y de la industria de fabricación que se extiende de la supervisión simple a la acción del lazo de control del lazo cerrado. Pruebas y pruebas de campo con varios dispositivos inalámbricos

han demostrado que la estabilidad, precisión, rendimiento y confiabilidad de la comunicación pueden satisfacer fácilmente las demandas de varias aplicaciones de monitoreo y control de procesos industriales. (Sreejith, Harikrishnan, Remya, & Sujith, 2014)

#### **2.8.1.1. Arquitectura WirelessHART.**

WirelessHART apunta a sensores y actuadores, equipos rotativos, como secadores de hornos, y aplicaciones de seguridad y salud ambiental, como duchas de seguridad, monitoreo de condiciones y fabricación flexible en la cual una parte de la planta puede ser reconfigurada para productos específicos. WirelessHART también impulsó extensiones al protocolo HART central, asegurando que los dispositivos más nuevos, como los monitores de vibración, fueran totalmente compatibles. La arquitectura estándar que normalmente se representa para WirelessHART se muestra en la figura 2.18.

Los tipos de dispositivos de red básicos incluyen:

- a. Dispositivos de campo (*Field Device*), realiza funciones de detección o activación de campo;
- b. Dispositivo enrutador (*Router Device*), todos los dispositivos deben tener la capacidad de enrutar los paquetes en una malla inalámbrica;
- c. Adaptadores (*Adapter*), se encarga de conectar dispositivos HART cableados a la red inalámbrica;
- d. Dispositivos de mano (*Handheld Device*), llevados por usuarios móviles como ingenieros y técnicos de planta;
- e. Puntos de acceso (*Access Point*), que permiten conectar a una malla inalámbrica a la puerta de enlace;
- f. Puerta de enlace (*Gateway*) simple o redundante, la misma funciona como un puente para las aplicaciones de host;
- g. Gestor de red (*Network Manager*) único, puede ser redundante y puede residir en dispositivos Gateway o estar separado de la puerta de enlace;

- h. Gestor de seguridad (*Security Manager*) que puede residir en el dispositivo de puerta de enlace o puerta.

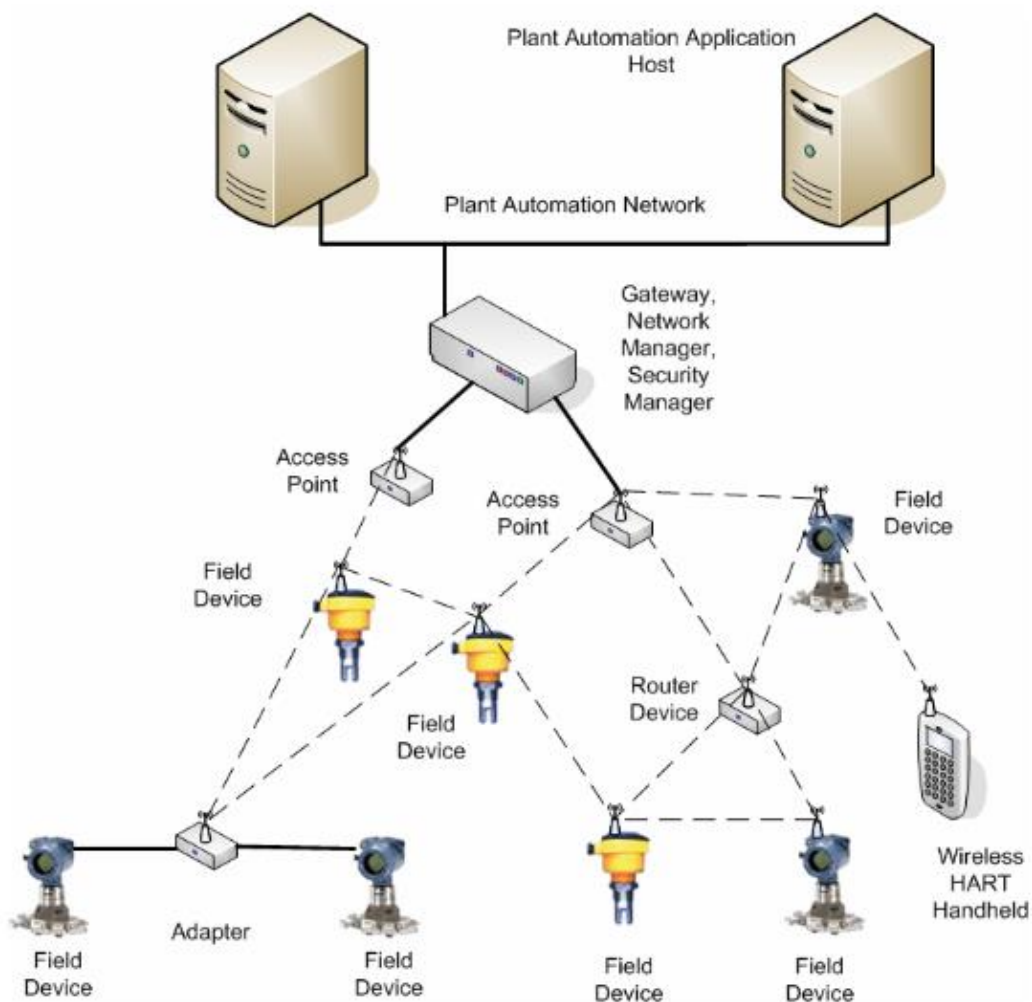


Figura 2. 18: Esquemático de la arquitectura WirelessHART.  
Fuente: (Nixon, 2012)

### 2.8.1.2. Capas de protocolos WirelessHART.

En la figura 2.19 se muestra la arquitectura de la capa de protocolos WirelessHART de acuerdo con el modelo de comunicación OSI de 7 capas. Se puede apreciar, que la pila de protocolos WirelessHART incluye cinco capas: (a) capa física, (b) capa de enlace de datos, (c) capa de red, (d) capa de transporte, y (e) capa de aplicación. (Song et al., 2008)

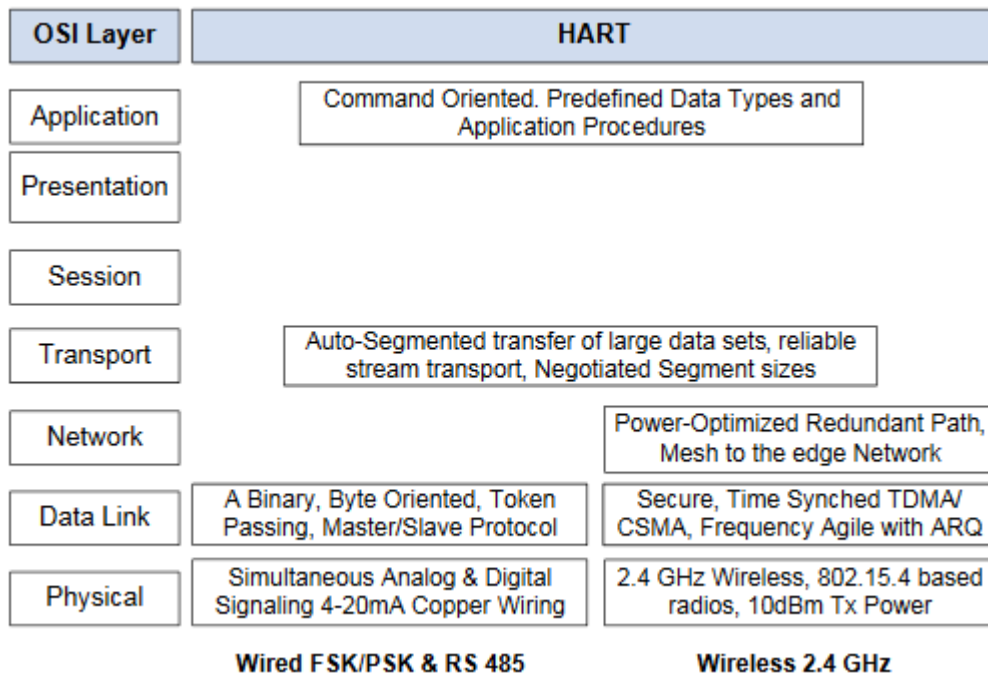


Figura 2. 19: Diagrama de las capas de WirelessHART.

Fuente: (Song et al., 2008)

### 2.8.1.3. Componentes de un sistema WirelessHART.

En la figura 2.20 se muestran los componentes básicos de una red WirelessHART. Para el caso de la estación de usuario, hace posible la conexión de una red WirelessHART de malla con el sistema de usuario conectado por cable; es decir el controlador de procesos automáticos y/o la aplicación de usuario. El controlador de procesos automáticos (por ejemplo, un PLC o DCS) conecta la estación de usuario mediante un bus de campo industrial (por ejemplo, Modbus, Profibus, etc.). El controlador de procesos automáticos usualmente accede a una “imagen” de las variables de los procesos de entrada y salida capturadas en la sección del “Gateway” del sistema de usuario. (Song et al., 2008)

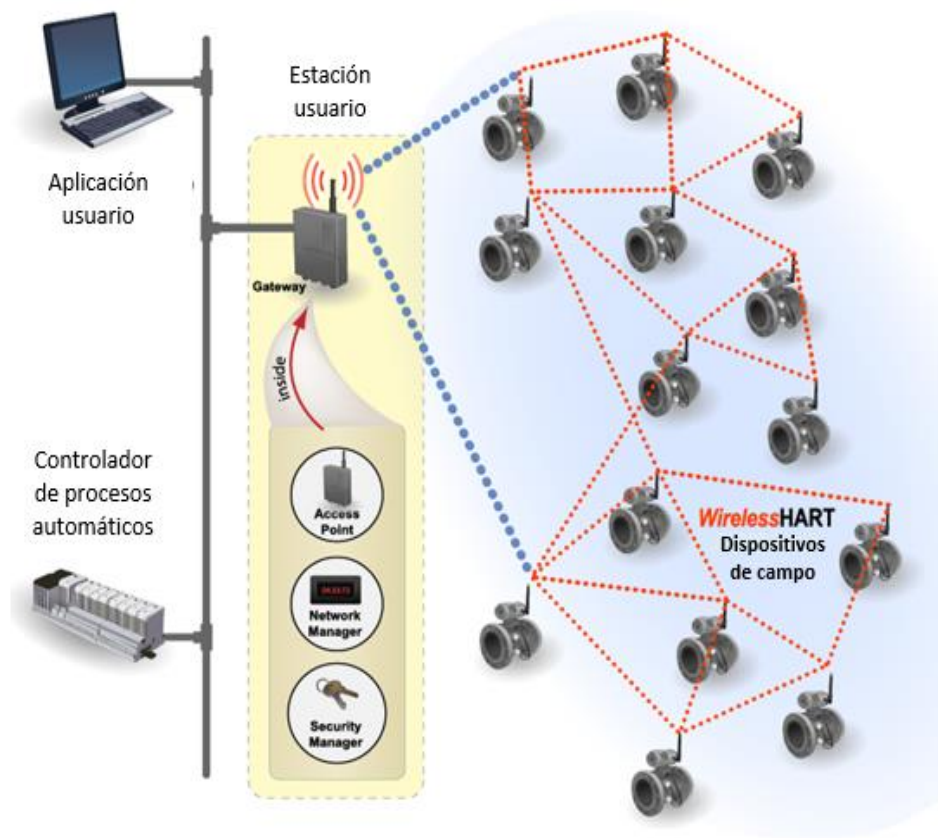


Figura 2. 20: Componentes de una red WirelessHART  
Fuente: (Lu, 2016)

### 2.8.2. Estándar ISA SP 100.

La Sociedad Internacional de Automatización (*International Society of Automation, ISA*) es una organización sin fines de lucro con sede en los Estados Unidos, compuesta por unos 20.000 profesionales de la automatización. ISA 100.11a-2009 (ISA SP100) se encarga de proveer las especificaciones para la realización de operaciones inalámbricas seguras y confiables en aplicaciones de monitoreo, alerta, control de calidad de lazo abierto y cerrado y monitoreo predictivo en ambientes industriales. (Oliva Alonso & Alonso Castro Gil, 2013)

Para Nixon, (2012) ISA 100.11a está destinado a ser parte de una familia de estándares diseñados para soportar una amplia gama de necesidades de plantas industriales inalámbricas, incluyendo automatización de procesos,

automatización de fábricas y RFID. Los criterios de diseño para ISA100.11a incluyen:

- a. flexibilidad,
- b. soporte para protocolos múltiples,
- c. uso de estándares abiertos,
- d. soporte para múltiples aplicaciones,
- e. fiabilidad (detección de errores, salto de canales),
- f. determinismo (TDMA, QOS apoyo),
- g. seguridad.

Para Colbert & Kott, (2016) ISA 100.11a describe una red de malla diseñada para proporcionar comunicación inalámbrica segura para procesar el control. Crea la capa de enlace de datos, la capa de red, la capa de transporte y la capa de aplicación (en la parte superior de la capa física de IEEE 802.15.4-2006, tal como se muestra en la tabla 2.2).

Tabla 2. 2: Capa de protocolo ISO 100.11a.

<b>Frecuencia</b>	<b>868,3 MHz</b>	<b>902-928 MHz</b>	<b>2400 a 2483,5 MHz</b>
<b>N° de Canales</b>	1	10	16
<b>Velocidad de datos (kbps)</b>	20	40	250
<b>Ubicación</b>	Europa	América	Mundial

Fuente: (Colbert & Kott, 2016)

ISA 100.11a admite dos tipos de topología de red: estrella y malla. ISA 100.11a tiene mecanismos de enrutamiento en dos niveles diferentes: (1) enrutamiento de malla de nivel de subred y (2) enrutamiento de nivel de backbone.

### Capítulo 3: Diseño para la implementación de una red de sensores inalámbricos en la industria petrolera.

Son muchos los criterios que facilitan la implementación de manera inalámbrica de los sensores que a menudo se utilizan en la industria petrolera, por lo que en este capítulo se realiza el análisis y procedimiento para su ejecución.

#### 3.1. Análisis de campo.

En el análisis de campo mediante medición se obtuvieron las distancias requeridas para cubrir la conectividad de los sensores. En la figura 3.1 se muestran los tanques de almacenamiento, donde serán colocados los nodos sensores de medición de nivel para su constante monitoreo.



Figura 3. 1: Tanques de almacenamiento de combustible.

Elaborado por: Autor.

Otro de los sitios que requiere de la instalación de sensores de presión son los tanques de almacenamiento de GLP, ya que con esto ayudaría a tener un almacenamiento y monitoreo eficiente. En la figura 3.2 se observa las ubicaciones y su geografía para poder realizar una correcta implementación de la red de sensores inalámbricos.



Figura 3. 2: Campo de tanques de almacenamiento de GLP.  
Elaborado por: Autor.

### **3.2. Elementos utilizados para la red inalámbrica de sensores.**

En esta sección se describen los elementos utilizados para realizar las pruebas de validación de gestión de la WSN. Los resultados obtenidos serán parte importante al momento de la implementación de un sistema de gestión web para monitorear los datos.

#### **3.2.1. Módulo ESP8266**

Para la conectividad de ciertos sensores se utilizaron los módulos inalámbricos ESP8266, por su facilidad de adquisición en el mercado nacional y por el costo fueron ideales para realizar la implementación a pequeña escala. En la figura 3.3 se muestra este elemento que es de gran utilidad inclusive para la tendencia actual como el internet de las cosas (*Internet of Things, IoT*) y que facilitarán la implementación de una red inalámbrica para la gestión mediante un software web de los sensores.

El módulo ESP8266 wifi es capaz de alojar una aplicación y distribuir cualquier resultado dentro de una red wifi ya que utiliza el protocolo TCP/IP, este se encuentra integrada al dispositivo en mención.





Figura 3. 3: Modulo Wifi ESP8266.  
Elaborado por: Autor.

Cada módulo ESP8266 viene preprogramado con un firmware de conjunto de comandos AT, lo que significa que puede conectarse a una placa embebida, tal como Arduino y con el mismo poder transmitir información a un servidor externo. A continuación, se enumera las características de estos elementos:

- Wifi direct (Peer to Peer, P2P)
- Interruptor TR integrado, LNA, amplificador de potencia.
- PLLs integrados.
- 1 MB de memoria flash.
- +19.5 dBm de potencia de salida en modo 802.11b.
- Protocolo TCP/IP integrada
- Consumo de energía en espera <math><1.0\text{mW}</math> (DTIM3).

Además, cuenta con una gran variedad de pines o puertos digitales y analógicos que pueden ser utilizados en lo que se requiera como se muestra en la figura 3.4. En este módulo se cargará un código fuente que responderá a la variación de voltaje por medio de un potenciómetro para simular la temperatura o presión como resultado de un sensor, por ejemplo, que se encuentre ubicado en un tanque de almacenamiento.

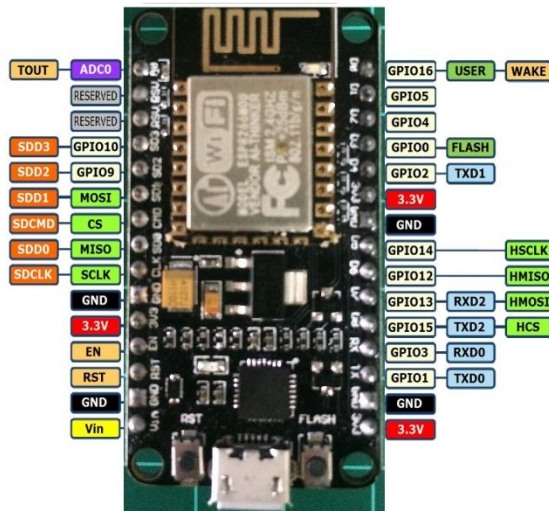


Figura 3. 4: Pinout ESP8266.  
Elaborado por: Autor.

### 3.2.2. Router D-link DIR-600

Para la red inalámbrica del presente trabajo se necesita un punto de acceso (*Access Point, AP*), el mismo que nos va permitir conectar varios puntos de red. Para esto se utiliza el router inalámbrico modelo DIR-600 de la marca D-link ya que por su bajo costo fue escogido para la implementación.



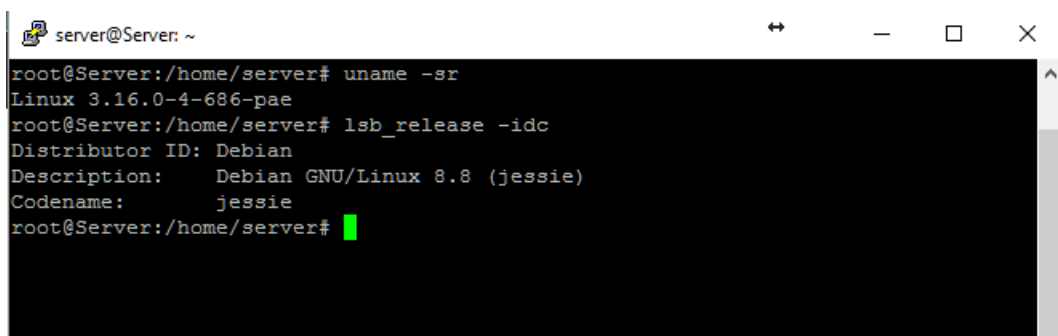
Figura 3. 5: Router D-Link.  
Elaborado por: Autor.

El router utiliza la tecnología Wireless 150 que ofrece velocidades superiores a los estándares 802.11b/g, además incluye un firewall de inspección de estado de paquete (SPI) que analiza el tráfico de red. Además, soporta cifrado WEP, WPA y WPA2 para mantener la red segura ante intrusos.

En la siguiente figura 3.5 se muestra el router utilizado para las pruebas e implementación.

### 3.2.3. CPU AMD Turbo 3600 i686

Para la implementación a pequeña escala del proyecto se utilizó una pequeña computadora con arquitectura i686 en el que se instaló el sistema operativo Debian 8. En la figura 3.6 se muestra los detalles del computador utilizando la consola del sistema operativo.

A screenshot of a terminal window titled 'server@Server: ~'. The terminal shows the following commands and output:

```
root@Server:/home/server# uname -sr
Linux 3.16.0-4-686-pae
root@Server:/home/server# lsb_release -idc
Distributor ID: Debian
Description:    Debian GNU/Linux 8.8 (jessie)
Codename:       jessie
root@Server:/home/server#
```

Figura 3. 6: Detalles de CPU utilizado como servidor debian.  
Elaborado por: Autor.

Entre las características del CPU están:

- Procesador AMD turbo 3600+x 1.7GHz
- Memoria RAM de 2 GBytes
- Disco duro de 160 GBytes
- Tarjeta de red VIA Technologies, Inc. VT6102 [Rhine-II]
- Unidad de CD incluida LG

### 3.3. Diseño de red inalámbrica.

En esta sección se detalla el diseño de la red que se utilizará para la implementación. Como se explicó en el capítulo 2, las comunicaciones inalámbricas opera con 3 tipos de topologías, tales como: (a) punto a punto, (b) punto a multipunto, y (c) multipunto a multipunto. En la implementación del presente trabajo, se utiliza la topología punto a multipunto ya que nos facilita la conectividad de varios dispositivos a un AP y posteriormente coordinar los resultados a un servidor.

Es importante recordar, que en una red inalámbrica el uso del espectro radioeléctrico para la transmisión de datos desde varios puntos, esta tecnología ha revolucionado en los últimos años determinando así varios estándares. El avance de los estudios que se realizan día a día en el campo de las redes inalámbrica nos ayuda a tener una respuesta más precisa en la transmisión de datos que generan cualquier dispositivo, además, en la actualidad se aplica en muchos dispositivos otorgando facilidad de conectividad y movilidad.

En la figura 3.7 se muestra la topología utilizada para la conectividad de los sensores y el servidor en nuestro router. Esta topología se aplica después de realizar un análisis de propagación en el sitio que se desea implementar, ya que puede existir interferencias el cual perjudicarían a la transmisión de resultados. Se puede ver las asignaciones de direcciones IP: (a) para el servidor Debian 192.168.100.16, y (b) de los cuatro nodos sensores ESP8266, que son 192.168.100.20 /21 /22 /23.

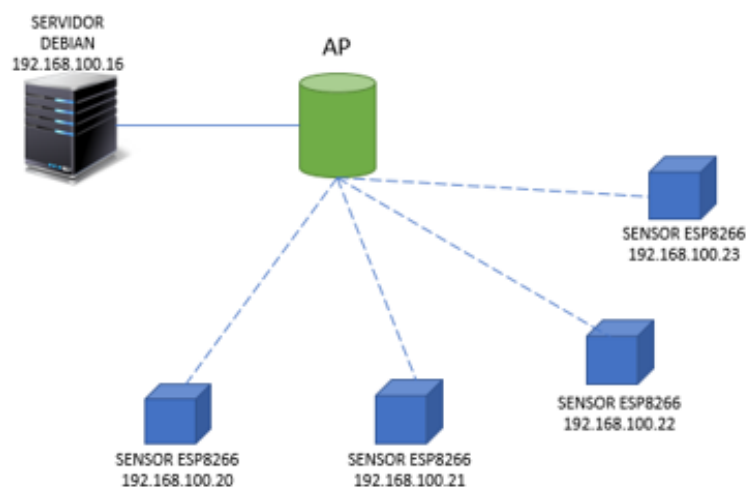


Figura 3. 7: Topología utilizada en la implementación de WSN.  
Elaborado por: Autor.

### 3.4. Diseño en ISIS Proteus y programación de los sensores ESP8266.

Como se indicó anteriormente, se utilizará los módulos ESP8266 para simular el funcionamiento de sensores de temperatura, presión, flujo y nivel, ya que son los datos más requeridos en la industria petrolera. Para ello es

necesario cargar un software en código de programación que nos ayude junto con un potenciómetro a realizar una variación de una señal analógica y convertirla en digital con un máximo de 12 bits.

En la figura 3.8 se presenta el diseño esquemático de la tarjeta Arduino nano (para ISIS Proteus está librería la llaman Simulino Nano) para simular por código la conversión de una señal analógica a digital. Se utiliza esta tarjeta Arduino nano para facilitar la simulación, debido a que los módulos inalámbricos ESP8266 no se encuentran en la librería del software ISIS Proteus 8.

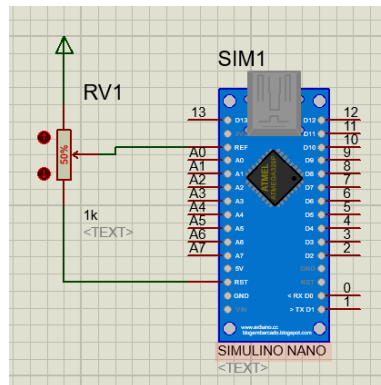


Figura 3. 8: Esquemático para la simulación de sensor de temperatura.  
Elaborado por: Autor.

Adicionalmente, vale hacer énfasis que el mismo código de programación que se utiliza para estas tarjetas embebidas serán cargadas en cada uno de los módulos ESP8266 activando unas librerías del IDE de Arduino. A continuación, se presentan por segmentos el código de programación en C++ aplicado en el módulo ESP8266 desde el IDE de Arduino. En la primera parte (véase la figura 3.9), se llama a las librerías ESP8266Wifi y pubsubclient que son responsables para especificar la tarjeta con la que se trabaja en la implementación y de realizar el intercambio de datos y negociación entre el módulo y el router wifi para permitir la conectividad a la red.

```

#include <ESP8266WiFi.h>
#include <PubSubClient.h>
const char* ssid = "ANetwork";
const char* password = "network1234";
const char* mqtt_server = "192.168.100.16";

float val = 0;
int analogPin = 0;

WiFiClient espClient;
PubSubClient client(espClient);
float temp = 0;

```

Figura 3. 9: Declaración de librerías y variables.  
Elaborado por: Autor.

Una vez llamadas las librerías mencionadas anteriormente, se procederá a crear las variables char con el contenido de los datos para la red wifi del presente trabajo, que para nuestro caso el SSID. SSID es el nombre de la red wifi, <<password>> es la clave para acceder y por último el <<mqtt\_server>> que es la dirección IP asignada a nuestro servidor.

El <<mqtt>> es un protocolo de comunicación utilizado en el internet de las cosas (IoT) que nos ayuda a comunicar diferentes dispositivos en una sola red y enviar datos leídos por sensores digitales o analógicos a un servidor para almacenarlos en una base de datos. Este protocolo se utiliza para la simulación e implementación del sistema de gestión, además puede ser sustituido por cualquier otro que se requiera.

Posteriormente se especifican las variables <<float val>>, este será el contenedor del valor analógico que se ingresa por el pin 0. Las siguientes líneas que indican <<Wificlient>> y <<PubSubClient>> son modos en los que trabajan las librerías llamadas en las primeras líneas del código mostrado en la figura 3.9, en este caso especificamos que el módulo será configurado como cliente. En la figura 3.10 se muestra el código que permite crear una función que realiza la negociación y anclaje a la red wifi con el router o AP. Con esto se logra la seguridad de la placa que ha sido anclada a una red wifi, el cual fue especificado en el código de programación (véase figura 3.10).

```

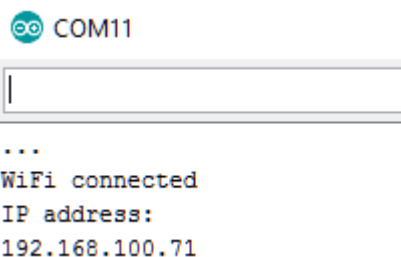
void setup_wifi() {
  delay(10);
  // Iniciamos la conectividad a la red wifi
  Serial.println();
  Serial.print("Conectando a ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    {
      delay(500);
      Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
  }
}

```

Figura 3. 10: Función para conexión de red inalámbrica wifi.  
Elaborado por: Autor.

Si conectamos el módulo a un puerto del PC este procede a enviar los datos del status por el puerto serial, con lo cual se puede visualizar los mensajes en el código de la función <<setup\_wifi>>. En la figura 3.11 se muestra el estado de la placa anclada en el puerto COM11 a un AP.



```

COM11
...
WiFi connected
IP address:
192.168.100.71

```

Figura 3. 11: Estado de conectividad a red inalámbrica wifi.  
Elaborado por: Autor.

Una vez conectado a la red wifi, es necesario buscar el servidor <<mqtt>> para poder comenzar a enviar los datos de los resultados a este, en el caso de que no sea posible conectar este, debe quedar en un ciclo intentándolo reiteradas veces como se muestra en la figura 3.12.

```

void reconnect() {
  // Conectar a servidor mqtt
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    if (client.connect("arduinoClient_temperature_sensor")) {
      Serial.println("connected");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // esperar 5 segundos hasta volver a intentar
      delay(5000);
    }
  }
}
}

```

Figura 3. 12: Función de conectividad a servidor mqtt.  
Elaborado por: Autor.

En la figura 3.13 se muestra la configuración de los pines que se utilizan en todo el código de programación. Es necesario inicializar el puerto serial para monitorear el estado del módulo en el terminal del puerto serial, así también inicia la función <<setup\_wifi>> para que inicialmente en el código se establezca una conexión entre el módulo y la red. Además, con la línea <<client-setServer>> especificamos la IP del servidor <<mqtt>> y del puerto que en este caso será 1883.

```

void setup()
{
  Serial.begin(115200);
  setup_wifi();
  client.setServer(mqtt_server, 1883);
}

```

Figura 3. 13: Función Setup de pines de la placa ESP8266.  
Elaborado por: Autor.

En la figura 3.14 se muestra el código principal que se encarga de tomar los datos del puerto ADC (convertidor A/D) pin 0 y transformarlos en digital, enviando al servidor como una variable string.



```

void loop()
{
  if (!client.connected()) {
    reconnect();
  }
  client.loop();

  val = analogRead(analogPin);
  val = map(val, 0, 1023, -50, 200);
  Serial.println(val);
  temp = val;
  delay(500);
  if((temp > -50) && (temp <200))
  {
    client.publish("ha/_temperature1", String(temp).c_str());
  }
}

```

Figura 3. 14: Función principal del código.  
Elaborado por: Autor.

En la primera parte de la función principal condicionamos el inicio, si el módulo se encuentra correctamente anclado a la red y al servidor, caso contrario deberá intentarlo nuevamente. Después, se asigna la lectura del puerto analógico a la variable `val` y con la función `<<map>>` se va a transformar el resultado a los valores entre -50 a 200 que corresponden a la temperatura. Posteriormente, se asigna el valor `<<val>>` a `<<temp>>` para enviarlo al servidor mediante la conectividad de cliente del módulo con la correcta sintaxis, tal como se observa en el código de la figura 3.14.

### 3.5. Implementación y configuración del servidor web

Para que el sistema de gestión sea funcional es necesario implementar un servidor que nos permita obtener y administrar los datos recibidos por cualquier sensor o periférico. Para esto fue necesario instalar en el servidor el sistema operativo Debian 8, ya que se tiene gran cantidad de información sobre su instalación y repositorios disponibles. No fue necesario instalar una interfaz gráfica, ya que una vez instalado el sistema operativo se puede administrar por `<<ssh>>` conectado remotamente con el programa llamado `<<Putty>>` el cual nos permite acceder desde cualquier punto de la red al

terminal de este y poder realizar las instalaciones y configuraciones necesarias. La figura 3.15 muestra la interfaz de la aplicación, pero que es suficiente para realizar cualquier tarea en la red.

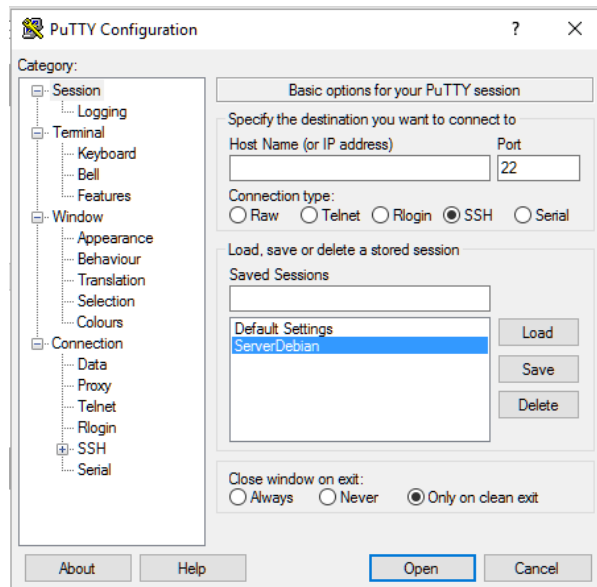


Figura 3. 155: Interfaz de Putty.

Elaborado por: Autor.

Una vez especificada la IP del servidor con el respectivo puerto podemos tener acceso al servidor e iniciar con las credenciales de usuario y contraseña, tal como se muestra en la figura 3.16.



Figura 3. 16: Ingreso al servidor via SSH.

Elaborado por: Autor.

Una vez inicializado el servidor se procede con la instalación y configuración del sistema de gestión web. Es recomendable, que antes de

iniciar cualquier tarea se tenga que actualizar el sistema, debido a que pueden existir repositorios descontinuados. La actualización se la puede conseguir escribiendo el siguiente comando <<apt-get update>>. Además, es necesario tener actualizado e instalado una lista de programas en el servidor como Python 2.7, Python-pip, Python-setuptools, MariaDB y librerías SSL.

Cuando se tiene listo el sistema operativo y los programas necesarios, se debe crear dos carpetas donde se va instalar el núcleo del sistema de gestión. Asignamos un usuario para administrar las carpetas, para que esto suceda se debe escribir las líneas de código que se muestran en la figura 3.17.

```
root@Server:/home/serverdebian# mkdir -p /opt/dmg
root@Server:/home/serverdebian# useadd -M domogik
bash: useadd: no se encontró la orden
root@Server:/home/serverdebian# useradd -M domogik
root@Server:/home/serverdebian# chown domogik /opt/dmg
root@Server:/home/serverdebian# chmod 777 /opt/dmg
root@Server:/home/serverdebian# █
```

Figura 3. 17: Procedimiento inicial de instalación del núcleo del sistema de gestión.  
Elaborado por: Autor.

Las líneas permiten crear una carpeta en el directorio <</opt>> con el nombre <<dmg>>, donde se descargan e instalan los dos paquetes necesarios para el desarrollo del sistema de gestión. En la figura 3.18 y 3.19 se presenta el procedimiento para descargar ambos paquetes de internet. Se descargan los paquetes <<domogik>> y <<domogik-mq>> que se guardan en la carpeta creada (véase la figura 3.17) para descomprimir e instalar. En general, estos paquetes serán el núcleo de todo el sistema, es decir, que se administra y configura en un entorno web los datos recibidos de los sensores instalados.

```

root@Server:/opt/dmg/domogik-mq# cd /opt/dmg/
root@Server:/opt/dmg# wget --content-disposition https://github.com/domogik/domogik/archive/0.5.2.tar.gz
--2017-06-22 20:59:04-- https://github.com/domogik/domogik/archive/0.5.2.tar.gz
Resolviendo github.com (github.com)... 192.30.253.112, 192.30.253.113
Conectando con github.com (github.com)[192.30.253.112]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Localización: https://codeload.github.com/domogik/domogik/tar.gz/0.5.2 [siguiendo]
--2017-06-22 20:59:04-- https://codeload.github.com/domogik/domogik/tar.gz/0.5.2
Resolviendo codeload.github.com (codeload.github.com)... 192.30.253.120, 192.30.253.121
Conectando con codeload.github.com (codeload.github.com)[192.30.253.120]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/x-gzip]
Grabando a: "domogik-0.5.2.tar.gz"

domogik-0.5.2.tar.gz      [          <=>          ] 26,57M  2,22MB/s  en 12s
2017-06-22 20:59:17 (2,13 MB/s) - "domogik-0.5.2.tar.gz" guardado [27862532]
root@Server:/opt/dmg# █

```

Figura 3. 18: Descarga de paquete Domogik desde github.  
Elaborado por: Autor.

```

root@Server:/opt/dmg# wget --content-disposition https://github.com/domogik/domogik-mq/archive/1.4.tar.gz
--2017-06-22 20:55:38-- https://github.com/domogik/domogik-mq/archive/1.4.tar.gz
Resolviendo github.com (github.com)... 192.30.253.112, 192.30.253.113
Conectando con github.com (github.com)[192.30.253.112]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Localización: https://codeload.github.com/domogik/domogik-mq/tar.gz/1.4 [siguiendo]
--2017-06-22 20:55:39-- https://codeload.github.com/domogik/domogik-mq/tar.gz/1.4
Resolviendo codeload.github.com (codeload.github.com)... 192.30.253.121, 192.30.253.120
Conectando con codeload.github.com (codeload.github.com)[192.30.253.121]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/x-gzip]
Grabando a: "domogik-mq-1.4.tar.gz"

domogik-mq-1.4.tar.gz    [ <=> ] 50,24K  299KB/s  en 0,2s
2017-06-22 20:55:39 (299 KB/s) - "domogik-mq-1.4.tar.gz" guardado [51444]
root@Server:/opt/dmg# █

```

Figura 3. 19: Descarga de paquete Domogik-mq desde github.  
Elaborado por: Autor.

Una vez obtenido ambos paquetes se procederá a su descompresión, instalación y configuración de las bases de datos, tal como se muestra en las figuras 3.20 y 3.21 las mismas son posicionadas en las carpetas correspondientes para ejecutar el archivo install.py.

```

root@Server:/home/server# cd /opt/dmg/domogik-mq
root@Server:/opt/dmg/domogik-mq# ./install.py --dist-packages █

```

Figura 3. 20: Comando para instalación de Domogik-mq.  
Elaborado por: Autor.

```

root@Server:~# cd /opt/dmg/domogik/
root@Server:/opt/dmg/domogik# ./install.py --dist-packages █

```

Figura 3. 21: Comando para instalación de Domogik.  
Elaborado por: Autor.

Posterior a la instalación se desplegará un mensaje (véase la figura 3.22) de confirmación, indicando que todo se ha procedido correctamente. Finalmente, el paquete domogik debe ser configurada la base de datos donde se van almacenar los resultados de las muestras obtenidas por los sensores. En la figura 3.23 se muestra en detalle la correspondiente configuración.

```
[ Check user config file contents ]
==> Config file correctly loaded
[ Parse [mq] section ]
[ Check ports availability ]
==> IPs/ports needed by Domogik MQ are not bound by anything else
==> [mq] section seems good
[ Check python version ]
==> Python version is >= 2.7

==> ===== <==
==> Everything seems ok <==
==> You can now install Domogik <==
==> ===== <==

root@Server:/opt/dmg/domogik-mq# █
```

Figura 3. 22: Mensaje de confirmación de correcta instalación de paquetes.  
Elaborado por: El Autor.

```
==> Directory /usr/sbin/ already exists
==> Copied file src/domogik/xpl/tools/64bit/xPL_Hub
[ Update /etc/default/domogik ]
[ Trying to find an interface to listen on ]
==> Selected interface eth0
[ Update the config file : /etc/domogik/domogik.cfg ]
[ Starting on section domogik ]
[ Starting on section database ]
- type [mysql]: mysql
- user [domogik]: domogik
- password [domopass]: domopass
- port [3306]:
- name [domogik]: domogik
- host [127.0.0.1]: 192.168.146.130
[ Starting on section admin ]
[ Starting on section butler ]
- name [Aria]: Aria
- lang [fr_FR]: en_US
- sex [female]: male █
```

Figura 3. 23: Creación de base de datos.  
Elaborado por: Autor.

Finalizado el proceso de instalación de los paquetes, es recomendable su actualización (véase la figura 3.24) para evitar problemas futuros con la configuración de los plugins requeridos.

```
root@Server:~# update-rc.d domogik defaults
```

Figura 3. 24: Actualización por defecto de domogik.  
Elaborado por: Autor.

Entre los paquetes requeridos para la ejecución del núcleo del sistema de gestión es necesario además instalar un servidor mqtt, más conocido como <<mosquitto>>. En línea se puede encontrar información referente a la instalación. Después de la instalación del paquete, se procede a ejecutarlo (véase la figura 3.25).

```
root@Server:~# /etc/init.d/mosquitto start
[ ok ] Starting mosquitto (via systemctl): mosquitto.service.
root@Server:~# /etc/init.d/mosquitto status
● mosquitto.service - Mosquitto MQTT Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; disabled)
   Active: active (running) since sáb 2005-01-01 21:30:40 -05; 18s ago
     Docs: man:mosquitto(8)
           https://mosquitto.org/
   Main PID: 2739 (mosquitto)
    CGroup: /system.slice/mosquitto.service
           └─2739 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

ene 01 21:30:40 Server systemd[1]: Started Mosquitto MQTT Broker.
ene 01 21:30:55 Server systemd[1]: Started Mosquitto MQTT Broker.
root@Server:~#
```

Figura 3. 25: Ejecución de servidor Mosquitto.  
Elaborado por: Autor.

Este servidor se encarga de recibir las muestras obtenidas por el convertidor ADC del módulo wifi ESP8266 para luego almacenarlos en la base de datos ya creada. Debido a que se trabaja con un sistema base para implementar el portal web de gestión, se debe instalar los plugins. Entre ellos está el client plugin-mqtt-server hasta domogik server (véase la figura 3.26). Este paquete es externo y abierto, podemos conseguir los detalles desde el github.

```
root@Server:~# dmz_package -i https://github.com/vdomos/domogik-plugin-script/archive/master.zip
```

Figura 3. 26: Instalación de plugin mqtt para sistema de gestión.  
Elaborado por: Autor.

Ahora, el sistema de gestión se encuentra con una base de datos lista para ser configurada, pero como se va trabajar con un portal web se requiere la instalación de un paquete llamado <<domoweb>>. Este permitirá la

administración y configuración del panel en el cual se reflejan de manera gráfica cada uno de los sensores, los cuales fueron previamente configurado. La figura 3.25 muestra las líneas de código en resumen que va permitir la obtención del paquete en línea y su instalación.

```
$ cd /opt/dmg
$ wget --content-disposition https://github.com/domogik/domoweb/archive/0.5.2.tar.gz
$ tar xvzf domoweb-0.5.2.tar.gz
$ ln -s domoweb-0.5.2 domoweb
$ cd domoweb
$ sudo ./install.py
```

Figura 3. 27: Instalación de paquetes requeridos para interfaz web.  
Elaborado por: Autor.

También, es necesario ejecutar los servicios y proceder a comprobar que todos ellos estén funcionando correctamente. Para lo cual es necesario que desde un PC a cualquier punto de la red se pueda acceder al portal <<backend>> y a <<frontend>> iniciando el navegador y direccionando a la IP del servidor. En la figura 3.28 se presenta el procedimiento de acceso al backend, lo cual es necesario para identificarse al sistema de administración con las credenciales creadas en la instalación de domogik.

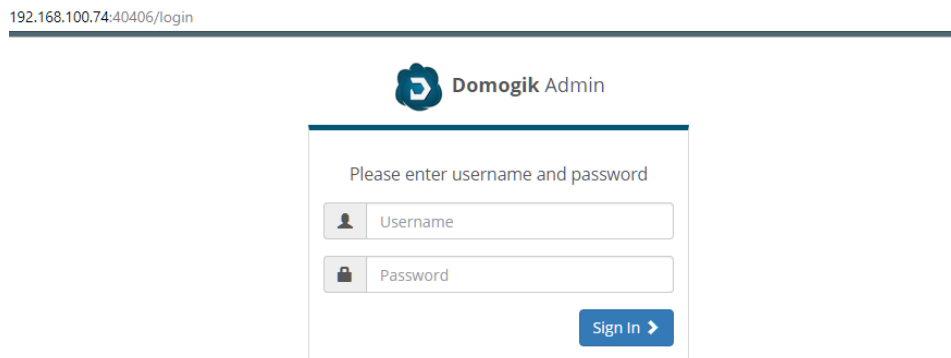


Figura 3. 28: Inicio de sesión para administrar desde el backend.  
Elaborado por: Autor.

Posterior al ingreso del backend (parte administrativa del portal web), se procede a comprobar que los plugins son mostrados a nodos sensores. Esta parte es necesaria habilitarla, si es posible el reseteo del servidor mosquito, y agregar los objetos que en este caso serán los sensores, tal como se muestra en las figuras 3.29 y 3.30.

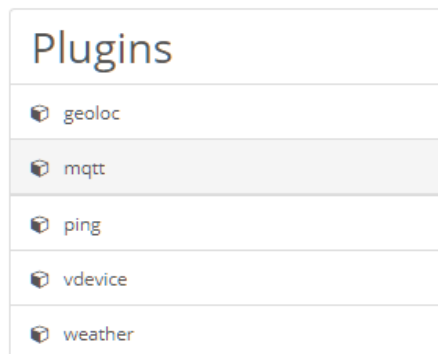


Figura 3. 29: Lista de plugin instalados en el sistema.  
Elaborado por: Autor.

Known Devices

- Temperature sensor
- Temperatura RF1
- arduinoClient\_temperature\_sensor

Temperatura RF1 mqtt.sensor\_temperature

Global parameters

ID	Name	Value	Type
8	qos	1	string
7	topic	ha/_temperature1	string

Figura 3. 30: Configuración de dispositivos clientes del servidor mqtt.  
Elaborado por: Autor.

Como es posible observar, en los parámetros globales del objeto se deben configurar, por ejemplo, el parámetro de QoS (en 1) y asignar una variable <<ha/\_temperature1>> como string. Del código de programación anterior del módulo wifi ESP8266, observamos que se asigna el mismo nombre al servidor mqtt, en este contenedor se almacenan los resultados de las muestras del sensor. Si se requiere crear nuevos objetos, es necesario ir al apartado del cliente mqtt e indicar el tipo de datos a recibir. En



consecuencia, la figura 3.31 presenta la pantalla de creación de un nuevo dispositivo.

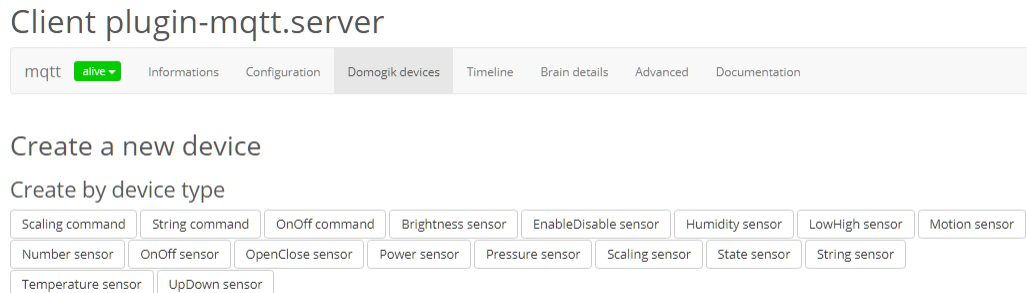


Figura 3. 31: Lista de dispositivos para implementación de obtención de resultados.  
Elaborado por: Autor.

Existen una limitada cantidad de tipos de datos que se pueden recibir en el servidor, y estos administrarlos adecuadamente con la base de datos. Además, es posible realizar un <<timeline>>, que sirve para mostrar los resultados recibidos, guardarlos y visualizarlos en la interfaz gráfica, tal como se muestra en la figura 3.32.

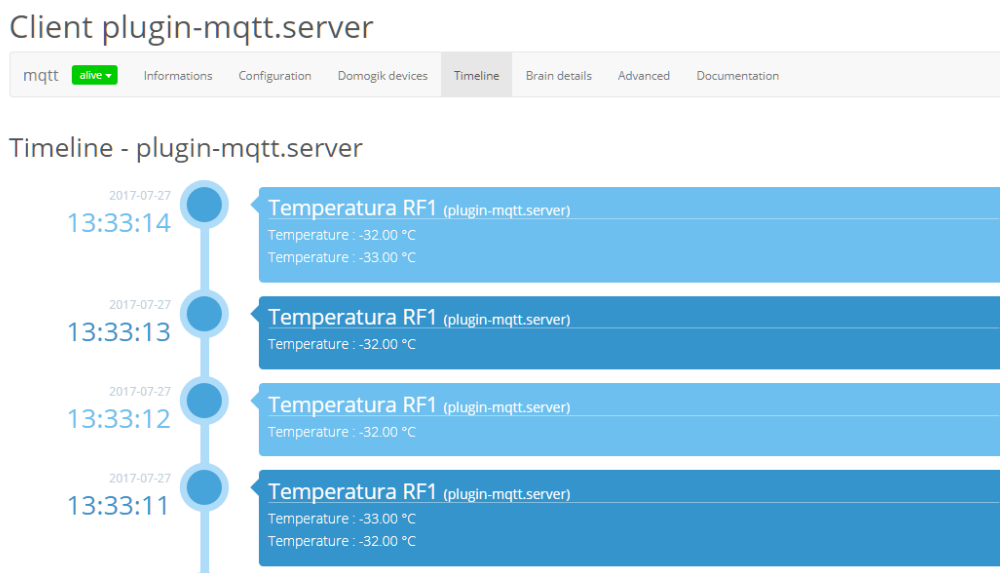


Figura 3. 32: Seguimiento de datos recibidos al servidor.  
Elaborado por: Autor.

A simple vista la interfaz de administración no es nada amigable hacia el usuario u operador de la central de monitoreo de la WSN. Es por esto que se configura el <<frontend>> del sistema de gestión, para lo cual escogemos el

navegador IP del servidor y el puerto 40404. En la figura 3.33 se muestra en pantalla el panel principal del sistema de gestión de la red inalámbrica de sensores (imagen de la empresa petrolera).



Figura 3. 33: Interfaz gráfica web del sistema de gestión.  
Elaborado por: Autor.

Vale reconocer los puntos más importantes del portal de gestión de WSN. En la esquina inferior izquierda se muestra el botón de paneles o secciones, es en ella que se agregan las pantallas necesarias para la gestión del centro de monitoreo, tal como se muestra en la figura 3.34.



Figura 3. 34: Secciones de interfaz web.  
Elaborado por: Autor.

De la figura 3.34 se divide en varias secciones, entre ellas están los sensores de temperatura, presión y flujo, además de una última sección llamada panel en el que se mostraran los resultados de los sensores más

críticos o de mayor importancia para la empresa. Por otro lado, tenemos en la esquina inferior derecha una herramienta que nos permite agregar widgets, configurar y agregar las secciones, tal como se muestra en la figura 3.35. Los widgets serán la representación gráfica de los datos obtenidos por los sensores. Sin embargo, la configuración será para modificar el entorno gráfico de la web y las secciones permitirán agregar o eliminar paneles que serán ventanas para mostrar distintos sensores clasificándolos como se requiera.



Figura 3. 35: Configuración de secciones y widgets.  
Elaborado por: Autor.

Los resultados de la configuración y despliegue de la interfaz gráfica web están a la vista en la figura 3.36, aunque se puede implementar cualquier tipo de señalización para representar n sensores. Además, se puede incluir cuadros estadísticos de los datos obtenidos en un determinado rango de tiempo que pueden variar entre semanas, días y horas, tal como se muestra en la figura 3.37.

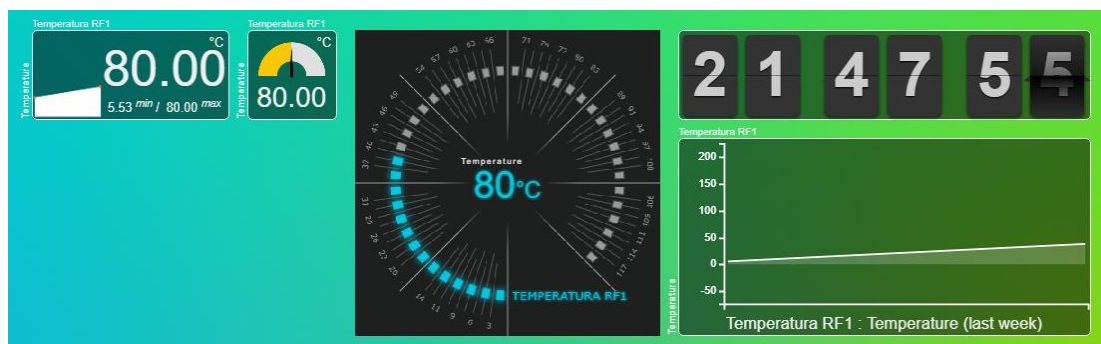


Figura 3. 36: Interfaz gráfica de resultados de sensor de temperatura simulada  
Elaborado por: Autor.

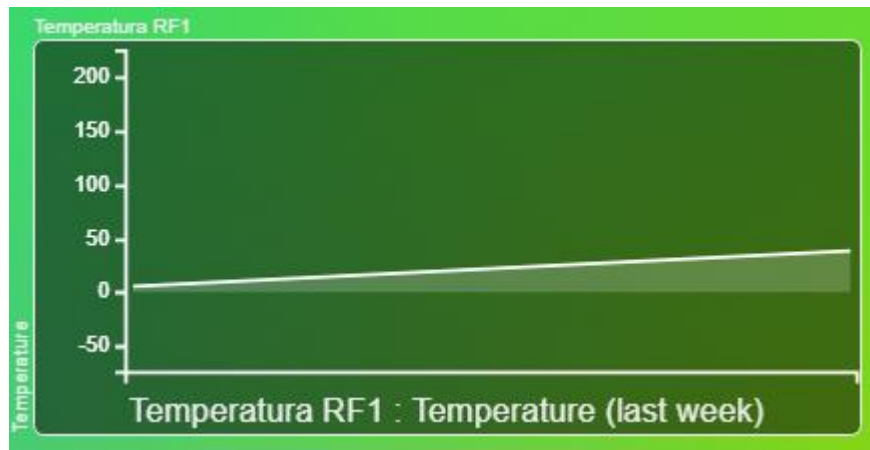


Figura 3. 37: Dato estadístico de las muestras tomadas en días.  
Elaborado por: Autor.

Además, es muy importante especificar qué tipo de sensor se utilizará en el campo ya que la interfaz necesita dicha información para especificar la unidad con la que estará trabajando, así como en la figura 3.38 muestra los datos obtenidos para el sensor de presión.

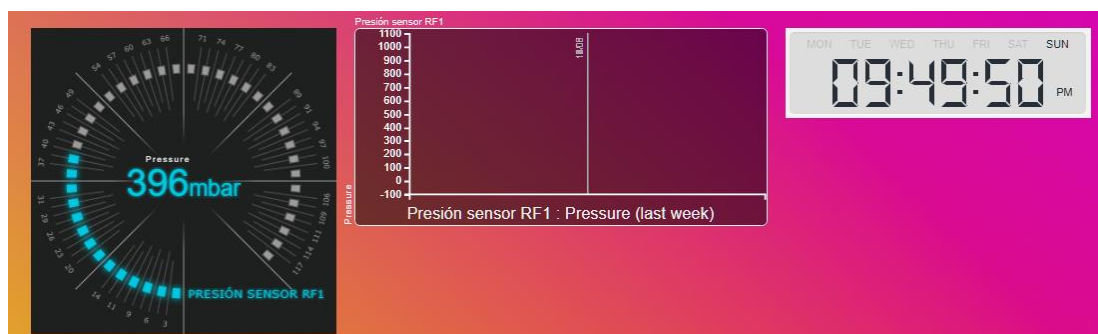


Figura 3. 38: Interfaz gráfica de resultados de sensor de presión.  
Elaborado por: Autor.

Así como en el panel de temperatura, este sistema gráfico web es capaz de acumular información por tiempo ilimitado y mostrarla en una tabla estadística como se ve en la figura 3.39.

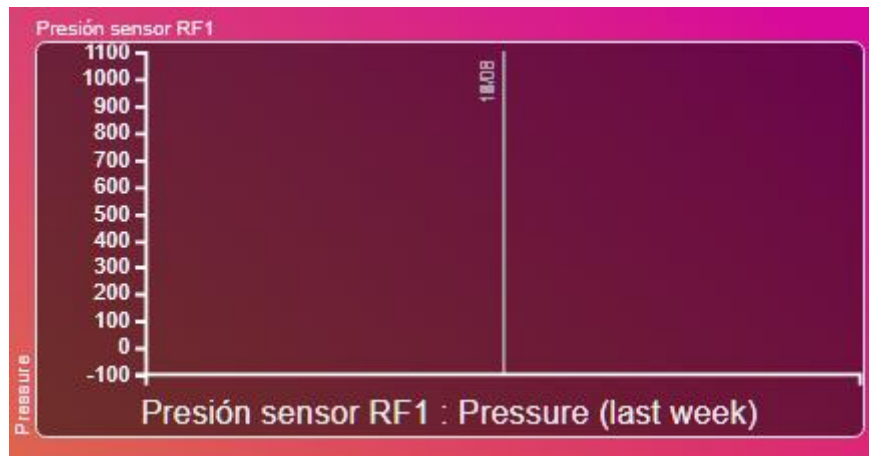


Figura 3. 39: Dato estadístico para sensor de presión.  
Elaborado por: Autor.

Finalmente, este sistema de gestión implementado a pequeña escala para su demostración del funcionamiento puede reducir costos y ser a la vez muy eficiente ya que el tiempo de respuesta de la transmisión de los datos al servidor es inmediata, si existe un problema de conectividad de estos a la red automáticamente nos indica una alarma de fallo de conexión.

## **Capítulo 4: Conclusiones y Recomendaciones.**

### **4.1. Conclusiones.**

1. La tecnología inalámbrica sigue avanzando en cuanto a aplicaciones, eliminando así la necesidad de cables, lo que permite implementaciones de redes eficientes en costos. Las redes de sensores inalámbricos pueden consistir en muchos tipos diferentes de sensores, cubriendo una amplia variedad de aplicaciones que consisten en Presión, Temperatura, Flujo, Nivel o simplemente retransmitir un cierre de contacto a través de un transmisor discreto.
2. La industria del petróleo y gas es quizás una de las industrias más frecuentes para la aplicación de redes de sensores inalámbricos. Sin embargo, estas aplicaciones suelen estar localizadas en áreas remotas, sin mencionar las exigencias ambientales. Independientemente, las redes de sensores inalámbricos se implementan para monitorear, administrar y controlar todo, desde tanques y compresores hasta generadores, separadores y cabezas de pozo. Las redes de sensores inalámbricos ofrecen ventajas sobre la tecnología cableada tradicional, donde las opciones con cable son demasiado caras o ni siquiera una opción
3. Los nodos sensores permitieron obtener los datos de medición de temperatura, presión y flujo, los cuales son monitoreados desde la aplicación web del sistema de gestión de sensores. Estos datos son almacenados para posterior reporte de información relacionada a la red de sensores inalámbricos.

## **4.2. Recomendaciones.**

1. Ampliar el estudio del presente trabajo de titulación, orientado a regulaciones ambientales con la finalidad de adaptarse a las iniciativas cada vez más reglamentarias en relación con las emisiones de gases de efecto invernadero, para lo cual una amplia gama de sensores inalámbricos se debe utilizar en las actividades de producción, procesamiento y transporte de petróleo y gas.
2. Implementar simulaciones de redes de sensores inalámbricos aplicando rutas de encaminamiento por estado del enlace usando el protocolo OSPF

## Bibliografía

- Agajo, J. (2016). ZigBee based Wireless patient temperature and pulse monitoring system. *Journal of Wireless Sensor Network*, 3(1), 1-11.
- Alaparathi, N., & Rao, P. S. (2016). Adaptation layer towards integration of ZigBee and IP stacks (pp. 212-215). IEEE. <https://doi.org/10.1109/ICACDOT.2016.7877581>
- Al-Karaki, J. N., & Kamal, A. E. (2004). Routing techniques in wireless sensor networks: a survey. *IEEE Wireless Communications*, 11(6), 6-28. <https://doi.org/10.1109/MWC.2004.1368893>
- Biddut, M. J. H., Islam, N., Sultana, R. S., Sarker, A., & Rahman, M. M. (2015). A new approach of ZigBee MAC layer design based on security enhancement (pp. 1-5). IEEE. <https://doi.org/10.1109/ICTP.2015.7427934>
- Cardell, J. B. (2017). ZigBee A Case Study. Recuperado 12 de junio de 2017, a partir de <http://www.science.smith.edu/~jcardell/>
- Carrión M., E., Maya O., E. A., Domínguez L., H. M., & Peluffo O., D. H. (2016). Sistema de Monitoreo de Monóxido de Carbono a través de una Red de sensores Inalámbrica y una Plataforma como servicio en la nube para una residencia. Recuperado 1 de junio de 2017, a partir de [https://www.researchgate.net/publication/311654601\\_Sistema\\_de\\_Monitoreo\\_de\\_Monoxido\\_de\\_Carbono\\_a\\_traves\\_de\\_una\\_Red\\_de\\_sensores\\_Inalambrica\\_y\\_una\\_Plataforma\\_como\\_servicio\\_en\\_la\\_nube\\_para\\_una\\_residencia](https://www.researchgate.net/publication/311654601_Sistema_de_Monitoreo_de_Monoxido_de_Carbono_a_traves_de_una_Red_de_sensores_Inalambrica_y_una_Plataforma_como_servicio_en_la_nube_para_una_residencia)
- Cedeño Villarroel, M. Á., & Eras Pérez, C. E. (2010, noviembre). *Diseño, implementación y monitoreo de una red Mesh/Zigbee aplicado a la*



*domótica, gestionada mediante SMS (Short Message Service), servidor web, correo electrónico, aplicación bluetooth para dispositivos móviles y control inalámbrico local.* Universidad Politécnica Salesiana. Recuperado a partir de <http://dspace.ups.edu.ec/handle/123456789/4383>

Colbert, E. J. M., & Kott, A. (Eds.). (2016). *Cyber-security of SCADA and other industrial control systems*. Switzerland: Springer.

Digi. (2017). XBee/XBee-PRO ZigBee RF Modules User Guide. Recuperado 3 de junio de 2017, a partir de <https://www.digi.com/resources/documentation/digidocs/PDFs/90000976.pdf>

Farnell. (2007). XBee Series 2 OEM RF Modules. Recuperado 3 de julio de 2017, a partir de <http://www.farnell.com/datasheets/27606.pdf>

Flores-Medina, M., Flores-García, F., Velasco-Martínez, V., González-Cervantes, G., & Jurado-Zamarripa, F. (2015). Monitoreo de humedad en suelo a través de red inalámbrica de sensores. *Tecnología y Ciencias del Agua*, VI(5), 75-88.

Gifty, J. J. D., & Sumathi, K. (2016). ZigBee Wireless Sensor Network simulation with various topologies (pp. 1-6). IEEE. <https://doi.org/10.1109/GET.2016.7916714>

Granda Bravo, B. A., & Belduma Belduma, L. A. (2017, abril). *Diseño de una red de sensores inalámbricos para el monitoreo del tránsito vehicular y la contaminación CO2 dentro de un sector urbano*. Universidad Politécnica Salesiana. Recuperado a partir de <http://dspace.ups.edu.ec/handle/123456789/14153>

- Hillman, M. (2017). An Overview of ZigBee Networks: A guide for implementers and security testers. MWR InfoSecurity. Recuperado a partir de <https://www.mwrinfosecurity.com/assets/Whitepapers/mwri-zigbee-overview-finalv2.pdf>
- Hobbytronics. (2017). XBee S2C 3mW module setup point-to-point. Recuperado 3 de junio de 2017, a partir de <http://www.hobbytronics.co.uk/xbee-s2c-configuration>
- Huang, X., Sharma, D., & Shah, P. G. (2011). Dynamic window with fuzzy controller in wireless sensor networks for elliptic curve cryptography (pp. 2342-2349). IEEE. <https://doi.org/10.1109/FUZZY.2011.6007327>
- IEEE. (2015). IEEE 802.15™: WIRELESS PERSONAL AREA NETWORKS (PANs). Recuperado 2 de julio de 2017, a partir de <http://standards.ieee.org/about/get/802/802.15.html>
- Iswandi, Nastiti, H. T., Praditya, I. E., & Mustika, I. W. (2016). Evaluation of XBee-Pro transmission range for Wireless Sensor Network's node under forested environments based on Received Signal Strength Indicator (RSSI) (pp. 56-60). IEEE. <https://doi.org/10.1109/ICSTC.2016.7877347>
- Kapitanova, K., Son, S. H., & Kang, K.-D. (2012). Using fuzzy logic for robust event detection in wireless sensor networks. *Ad Hoc Networks*, 10(4), 709-722. <https://doi.org/10.1016/j.adhoc.2011.06.008>
- Khalaf, A. A. M., & Mokadem, M. S. A. (2016). Effects of ZigBee component failure on the WSN performance with different topologies (pp. 9-12). IEEE. <https://doi.org/10.1109/ICM.2016.7847894>

- Kim, A. N., Hekland, F., Petersen, S., & Doyle, P. (2008). When HART goes wireless: Understanding and implementing the WirelessHART standard (pp. 899-907). IEEE. <https://doi.org/10.1109/ETFA.2008.4638503>
- Lu, C. (2016). WirelessHART: Applications in Process Industry. Recuperado 5 de julio de 2017, a partir de <http://www.cse.wustl.edu/~lu/>
- Mbowe, J. E., & Oreku, G. S. (2014). Quality of Service in Wireless Sensor Networks. *Wireless Sensor Network, 2014*. <https://doi.org/10.4236/wsn.2014.62003>
- Molina Arribere, G. (2017). Optimization for Sensor Networks [Research]. Recuperado 4 de agosto de 2017, a partir de [http://neo.lcc.uma.es/staff/guillermo/index\\_files/investigacion/redes%20de%20sensores.htm](http://neo.lcc.uma.es/staff/guillermo/index_files/investigacion/redes%20de%20sensores.htm)
- Nixon, M. (2012). A comparison of WirelessHART and ISA100 11a. Recuperado 5 de julio de 2017, a partir de <https://www.controlglobal.com/assets/12WPpdf/120904-emerson-wirelesshart-isa.pdf>
- Obodoeze, F., Inyama, H., & Idigo, V. (2012). Wireless Sensor Network in Niger Delta oil and gas field monitoring: The security challenges and countermeasures. *International Journal of Distributed and Parallel Systems (IJDPS)*, 3(6), 65-77. <https://doi.org/10.5121/ijdp.2012.3606>
- Oliva Alonso, N., & Alonso Castro Gil, M. (2013). *Redes de comunicaciones industriales*. UNED. Recuperado a partir de <http://site.ebrary.com/lib/interpuertoricosp/Doc?id=10804413>
- Ortega-Corral, C., Palafox, L. E., García-Macías, J. A., Sánchez-García, J., & Aguilar, L. (2014). End-to-End Message Exchange in a Deployable

Marine Environment Hierarchical Wireless Sensor Network. *International Journal of Distributed Sensor Networks*, 10(1), 950973. <https://doi.org/10.1155/2014/950973>

Rao, Y. C., Santhi, C., Professor, R., & Lavanya, P. (2012). Monitoring and Protection of Oil and Gas Condition in Industrial Using Wireless Sensor Networks. *International Journal of Electronics Communication and Computer Technology*, 2(5).

RF Wireless World. (2017). ZigBee mesh network tutorial: ZigBee mesh network basics. Recuperado 2 de junio de 2017, a partir de <http://www.rfwireless-world.com/Tutorials/Zigbee-mesh-network-tutorial.html>

Ruiz Sánchez, W. R. (2016). *Redes de sensores inalámbricos enfocadas a la medicina con énfasis en control de los signos vitales en pacientes adultos mayores*. Pontificia Universidad Católica del Ecuador. Recuperado a partir de <http://repositorio.puce.edu.ec:80/xmlui/handle/22000/11311>

Shue, S., Johnson, L. E., & Conrad, J. M. (2017). Utilization of XBee ZigBee modules and MATLAB for RSSI localization applications (pp. 1-6). IEEE. <https://doi.org/10.1109/SECON.2017.7925305>

Song, J., Han, S., Mok, A., Chen, D., Lucas, M., Nixon, M., & Pratt, W. (2008). WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control (pp. 377-386). IEEE. <https://doi.org/10.1109/RTAS.2008.15>

Sreejith, M., Harikrishnan, D., Remya, G., & Sujith, M. (2014). WirelessHART Communication based protocol for petrochemical complex.

*International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC)*, 2(6), 1607-1617.

Sun, B., Ahmed, F., Sun, F., Qian, Q., & Xiao, Y. (2016). Water quality monitoring using STORM 3 Data Loggers and a wireless sensor network. *International Journal of Sensor Networks*, 20(1), 26. <https://doi.org/10.1504/IJSNET.2016.074270>

Wang, X. hua, Jiao, Y. lin, Yang, J., & Niu, Y. chao. (2016). The acoustic emission detection and localization technology of the pipeline crack. *International Journal of Sensor Networks*, 20(2), 111. <https://doi.org/10.1504/IJSNET.2016.074700>

XBee. (2017). ¿Qué es XBee? Recuperado 3 de julio de 2017, a partir de <http://xbee.cl/que-es-xbee/>

Zhang, S., Yan, A., & Ma, T. (2013). Energy-Balanced Routing for Maximizing Network Lifetime in WirelessHART. *International Journal of Distributed Sensor Networks*, 9(10), 173185. <https://doi.org/10.1155/2013/173185>

Zhao, J., Li, M., Zhang, W., & Zheng, R. (2010). ECG signal adaptive filtering and QRS complex detecting method (pp. 869-872). IEEE. <https://doi.org/10.1109/BMEI.2010.5639903>

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Dender Zurita, Roberto Carlos** con C.C: # 080255089-7 autor del trabajo de titulación: Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera, previo a la obtención del título de **Magister en Telecomunicaciones** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 20 de Octubre de 2017

f. \_\_\_\_\_

Nombre: **Dender Zurita, Roberto Carlos**

C.C: 080255089-7

<b>REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA</b>		
<b>FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN</b>		
<b>TÍTULO Y SUBTÍTULO:</b>	Gestión de redes de sensores inalámbricos (WSN) para la industria petrolera	
<b>AUTOR(ES)</b>	Dender Zurita, Roberto Carlos	
<b>REVISOR(ES)/TUTOR(ES)</b>	MSc. Zamora Cedeño, Néstor Armando	
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil	
<b>FACULTAD:</b>	Sistema de Posgrado	
<b>PROGRAMA:</b>	Maestría en Telecomunicaciones	
<b>TITULO OBTENIDO:</b>	Magister en Telecomunicaciones	
<b>FECHA DE PUBLICACIÓN:</b>	20 de Octubre de 2017	<b>No. DE PÁGINAS:</b> 85
<b>ÁREAS TEMÁTICAS:</b>	Comunicaciones Inalámbricas, Diseño Electrónico Digital, Gestión de Redes.	
<b>PALABRAS CLAVES/ KEYWORDS:</b>	ZigBee, Sensores, Redes, Datos, Inalámbrico, WSN	
<b>RESUMEN/ABSTRACT (150-250 palabras):</b>		
<p>Para el presente trabajo de titulación se describieron los conceptos sobre la red de sensores inalámbricos, su composición y arquitectura. Así, como también los estándares de comunicación inalámbrico y su implementación sobre el estándar IEEE 802.15. Las redes de sensores inalámbricos (WSN) hoy en día son de gran ayuda por lo que permiten llegar a lugares dificultosos ya sean estos de gran altura o lugares confinados lo que facilita su acceso a la obtención de datos de una manera fácil, confiable y en línea sin necesidad de infraestructuras físicas. Posteriormente, se implementó una red de sensores inalámbricos que permita la expansión del sistema de una manera rápida debido a que existen múltiples aplicaciones para la obtención y procesamiento de datos. Después, se enfocó en la creación de un sistema de gestión de la red inalámbrica de sensores a escala para el modelado de una red de sensores inalámbricos en la industria petrolera. Los resultados obtenidos permitieron obtener mediciones de los eventos ocurridos en cada uno de los sensores que fueron evaluados en la WSN.</p>		
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO
<b>CONTACTO CON AUTOR/ES:</b>	<b>Teléfono:</b> 0997686865	<b>E-mail:</b> rd_yanio@hotmail.com
<b>CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):</b>	<b>Nombre:</b> Manuel Romero Paz	
	<b>Teléfono:</b> 0994606932	
	<b>E-mail:</b> <a href="mailto:mromeropaz@yahoo.com">mromeropaz@yahoo.com</a>	
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>		
<b>Nº. DE REGISTRO (en base a datos):</b>		
<b>Nº. DE CLASIFICACIÓN:</b>		
<b>DIRECCIÓN URL (tesis en la web):</b>		