

PROTOTIPO DE UN MÓDULO DE COMUNICACIÓN INALÁMBRICO PARA NODOS
COORDINADORES DE REDES DE SENSORES CON CONEXIÓN DIRECTA A UN
CENTRO DE FUSIÓN.

Adolfo Andrés Castro Sánchez



UNIVERSIDAD PONTIFICIA BOLIVARIANA
POSTGRADOS ESCUELA DE INGENIERÍAS
MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN
ÉNFASIS TELECOMUNICACIONES
MEDELLÍN
2015

PROTOTIPO DE UN MÓDULO DE COMUNICACIÓN INALÁMBRICO PARA NODOS
COORDINADORES DE REDES DE SENSORES CON CONEXIÓN DIRECTA A UN
CENTRO DE FUSIÓN.

Adolfo Andrés Castro Sánchez

Trabajo de grado para optar al título de Magister en Tecnologías de la Información y
Comunicación con Énfasis Telecomunicaciones

Director
Leonardo Betancur A, PhD



UNIVERSIDAD PONTIFICIA BOLIVARIANA
POSTGRADOS ESCUELA DE INGENIERÍAS
MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN
ÉNFASIS TELECOMUNICACIONES
MEDELLÍN
2015

Fecha: 31 de Julio de 2015

Nombre: Adolfo Andrés Castro Sánchez

“Declaro que esta tesis (o trabajo de grado) no ha sido presentada para optar a un título, ya sea en igual forma o con variaciones, en esta o cualquier otra universidad” Art 82 Régimen Discente de Formación Avanzada.

Firma

*"A Dios
mis padres
Clara y Tulio
y mis hermanos
Gustavo, Luis y Julián"*

Índice general

Indice General	III
Indice de Figuras	V
Indice de Tablas	VI
Resumen	VII
Planteamiento del Problema	VIII
Objetivos	X
1. Estado del Arte	1
1.1. Casos de estudio	2
1.1.1. Proyecto “Open WSN”	2
1.1.2. Caso Libelium	4
1.1.3. Otros Casos de Estudios	8
1.1.4. Proyectos relacionados con la problemática identificada.	10
1.2. Protocolos de comunicación para WSN.	11
1.2.1. Protocolo Zigbee y la “ <i>Alliance Zigbee</i> ”	12
1.2.2. Módulos de comunicación Inalámbricos de Digi.	14
1.2.3. Esquema de un módulo Xbee y una red Zigbee.	14
1.3. Ordenadores de Placa Reducida.	15
1.4. Aspectos del Problema	15
2. Diseño	18
2.1. Ordenador “Raspberry PI model B”	19
2.2. Módulos inalámbricos con protocolo Zigbee.	21
2.3. Descripción de los microcontroladores PIC	24
2.4. Diseño del prototipo de Hardware y Software	27
2.5. Diseño del Transmisor	28
2.6. Diseño del Receptor	29

2.7. Diagrama de Contexto y Casos de Uso.	30
2.7.1. Casos de uso	30
2.8. Modelo Entidad Relación (MER).	33
3. Dimensionamiento	34
3.1. Cálculo “Link Budget”.	34
3.2. Modelo del Canal	37
3.3. Sincronización de los nodos	40
3.4. Arquitectura de red para el dimensionamiento	41
3.5. Dimensión general de una red Zigbee	42
3.6. Enlace inalámbrico para el dimensionamiento	42
3.7. Simulación y análisis de una red Zigbee	43
3.7.1. Resultados de la simulación	44
3.8. Propuesta para el dimensionamiento de la red.	48
4. Implementación	50
4.1. Implementación del diseño de hardware	51
4.1.1. Esquema de Implementación	51
4.1.2. Implementación del Diseño.	51
4.1.3. Implementación por bloques	52
4.1.4. Implementación del Nodo Coordinador	54
4.1.5. Implementación del Nodo Sensor	55
4.2. Implementación del dispositivo	56
4.3. Niveles o capas del sistema implementado.	60
5. Centro de Fusión	61
5.1. Plataformas que guardan datos para WSN	62
5.1.1. “Cloud Device”	62
5.1.2. Plataforma “Xively”	63
5.1.3. ODK Sensor	64
5.2. Centro de Fusión	65
5.3. Flujo de información	67
5.4. Velocidad del enlace al centro de fusión	67
6. Análisis, Recomendaciones y Trabajos Futuros	73
6.1. Análisis de Escalabilidad y Latencia en una WSN	75
6.2. Recomendaciones	79
6.3. Trabajos Futuros	80
7. Conclusiones	82

Índice de figuras

1.1.	Arquitectura del proyecto Open WSN.	2
1.2.	Pila de protocolos del estándar “ <i>Open WSN</i> ”	3
1.3.	Waspote Plug&sentido!.	6
1.4.	Meshlium.	6
1.5.	Productos de la Alliance.	12
1.6.	Clasificación de las redes Inalámbricas.	13
1.7.	Componentes de una red WiFi y Zigbee.	13
2.1.	Raspberry Pi Modelo B.	19
2.2.	Esquema de pines de la Raspberry.	20
2.3.	Diagrama de pines de un Módulo de Xbee serie 2.	21
2.4.	Conexión Xbee – Microcontrolador.	22
2.5.	Esquema general de una trama API.	23
2.6.	Esquema básico de un microcontrolador.	24
2.7.	Diagrama de pines del PIC16F1837 encapsulado SIP.	25
2.8.	Bloque físico para programar un Microcontrolador.	26
2.9.	Diseño del prototipo.	27
2.10.	Diseño del Trasmisor.	29
2.11.	Diseño del Receptor.	29
2.12.	Diagrama Caso de Uso 1 “Ejecutar Script”.	30
2.13.	Diagrama Caso de Uso 2 “Visualizar los datos”.	31
2.14.	Modelo Entidad Relacion de la base de datos.	33
3.1.	Descripción del Link Budget.	35
3.2.	Grafica pérdidas de trayectoria en un escenario abierto.	38
3.3.	Perdidas por trayectoria.	39
3.4.	Radio de cobertura para una topología en estrella.	39
3.5.	Topologías de red contempladas en la norma IEEE 802.15.4.	41
3.6.	Composición de una trama física en el estándar 802.15.4	42
3.7.	Grafica carga de la Red Vs # Nodos	45
3.8.	Grafica Delay end to end Vs # Nodos.	45

3.9. Grafica tráfico recibido Nodo Coordinador.	46
3.10. Grafica Throughput Vs Nodos.	46
3.11. Gráfica Tiempo de transmisión de los paquetes Vs # Nodos.	48
3.12. Grafica de velocidad 170Kbps Vs # Nodos.	49
4.1. Diseño general normalizado	51
4.2. Diseño general implementado en pruebas.	52
4.3. Estructura por bloques del nodo sensor.	52
4.4. Estructura por bloques del nodo coordinador.	53
4.5. Estructura por bloques del centro de fusión.	53
4.6. Diseño normalizado del nodo coordinador.	54
4.7. Diseño implementado del nodo coordinador.	54
4.8. Diseño normalizado nodo sensor.	55
4.9. Diseño implementado en pruebas del nodo sensor.	55
4.10. Diagrama de Flujo del algoritmo en el Micro	56
4.11. Algoritmo en PHP para la recepción de los datos	58
5.1. Plataforma device cloud	62
5.2. Elementos para subir los datos de un Xbee a la nube a través de Cloud.	63
5.3. Esquema general de conexión a Xively.	63
5.4. Flujo de los datos hacia Xively.	64
5.5. Aplicación ODK sensor para Android.	65
5.6. Centro de Fusión.	66
5.7. Flujo de Información.	67
5.8. Tráfico de datos y conexión a la nube.	68
5.9. Velocidad de enlace para variables lentas (temperatura).	72
6.1. Tráfico de datos en una WSN tipo “Mesh”.	74
6.2. Pequeñas redes de sensores tipo Estrella.	75
6.3. Escenarios de escalabilidad.	76
6.4. Un nodo coordinador con 100 nodos sensores.	77
6.5. Un nodo coordinador con 10 nodos sensores.	77
6.6. Aspectos del primer escenario.	78
6.7. Aspectos del segundo escenario.	79

Índice de tablas

1.1.	Plataformas comerciales para desarrollo y diseño digital	5
1.2.	Algunos proyectos desarrollados por Libelium.	7
1.3.	La serie Xbee de Digi internacional.	14
1.4.	Comparación entre los elementos de una red Zigbee y una red Digi.	15
1.5.	Aspectos o criterios de Diseño.	16
2.1.	Diseño del prototipo por bloques.	18
2.2.	Ventajas y desventajas de utilizar Raspberry Pi modelo B.	20
2.3.	Ventajas y desventajas de utilizar comandos AT.	22
2.4.	Ventajas y desventajas de utilizar tramas API.	22
2.5.	Ventajas y desventajas de utilizar los Módulos Xbee serie 2.	23
2.6.	Pines del PIC16F1937 utilizados en diseño.	25
2.7.	Ventajas y desventajas de utilizar el microcontrolador PIC16F1937.	26
2.8.	Descripción del caso de uso número 1.	31
2.9.	Descripción del caso de uso número 2.	32
3.1.	Parámetros de las antenas Xbee S2 Digi	36
3.2.	Características de los dispositivos Zigbee utilizados.	38
3.3.	Parámetros de simulación Topología en estrella.	44
3.4.	Resultado de la simulación en diferentes escenarios.	44
3.5.	Tiempo de transmisión de los paquetes.	47
3.6.	Criterio para el dimensionamiento de una red Zigbee (Estrella).	48
4.1.	Niveles o capas de la aplicación realizada.	60
5.1.	Asignación de valores a las variables.	69
5.2.	Trama Zigbee + SQL.	69

Resumen

En la actualidad, la gestión del riesgo ambiental forma parte de la administración de una ciudad por lo tanto la instalación de redes de sensores se convierte en la plataforma tecnológica que recolecta la información del medio ambiente necesaria para gestionar “atención y prevención de desastres”, Para ello la mayoría de los gobiernos públicos para ciudades ubicuas utilizan estas redes de sensores, para la automatización y monitorización de áreas con alto riesgo de desastres naturales mejorando los procesos de toma de decisiones ante un suceso imprevisto y disponiendo de servicios en tiempo real [1]. En la gestión del riesgo se definen parámetros críticos a la hora de un diseño de red de sensores, en los que hay que tener en cuenta problemas de: escalabilidad, latencia, retardos en la entrega de paquetes, plataforma de envío de datos, heterogeneidad de redes, sincronización de datos entre muchos otros. Esta propuesta pretende diseñar un sistema de hardware que pueda ser implementado y que logre atacar en forma distribuida los problemas de escalabilidad y latencia. Dicho sistema funcionará dentro de una red inalámbrica de sensores que contiene aplicaciones de alto nivel para cargar datos directos en un servidor en internet. En el presente proyecto se abordará el problema de comunicación que permite atacar o aportar una solución a la dificultad producida por la congestión de datos en redes de sensores de gran cantidad de nodos a través de la alternativa del diseño de una arquitectura de red, para ello ésta será dividida en pequeñas redes de acceso distribuido con topología en estrella. En este escenario, cada una de las redes distribuidas requiere un acceso directo al centro de fusión, en donde la menor cantidad de nodos intermediarios mejora el desempeño del sistema lo cual se tratará y analizará con la implementación del dispositivo y definición de arquitectura de red para el funcionamiento del mismo.

Planteamiento del Problema

Una red de sensores, reúne un conjunto de dispositivos electrónicos interconectados entre sí, los cuales representan una solución técnica a la gestión del riesgo en ciudades inteligentes. Según L. Filipponi [2], la implementación de estas redes sugiere la topología de red en árbol, que por sus características naturales, posee problemas debido a la concentración de datos en algunos puntos; como consecuencia, cuando la red es puesta en funcionamiento, tiende por momentos a no reaccionar ante el envío simultáneo de información, saturando algunos puntos de la red y alterando el normal funcionamiento. De cualquier modo en espacios inteligentes los datos enviados a través de redes de sensores a diferentes dispositivos y aplicaciones produce grandes cantidades de información en tiempo real las cuales son procesadas por muchos dispositivos de hardware. En los puntos de mayor concentración de datos, la red puede perder calidad especialmente en nodos “Sink” o nodos colectores en donde se presenta la mayor concentración de datos recibidos o procedentes de diferentes puntos de red, en estos nodos se suelen formar “cuellos de botella” haciendo que la red pierda escalabilidad debido al retraso que se puede generarse. No obstante estos nodos colectores no solo reciben información sino que también tienen que enviarla al siguiente punto de red el cual ofrece un servicio, de acuerdo a las exigencias de tráfico de datos o aumento de trabajo, afectando su escalabilidad, en ciudades con grandes y exigentes recolección y monitoreo de datos procedente de la red de sensores sigue siendo una tarea difícil debido a las limitaciones de una WSN: ancho de banda, alto consumo energético debido a sus características de procesamiento así como el retardo en la entrega de los paquetes [3]. La latencia en la transmisión impide que los paquetes sean entregados en tiempos adecuados, por lo cual la información recolectada en diferentes puntos de la red implementada debe ser procesada para hacer correlación de los datos obtenidos sin embargo esta función podría ser compleja [4]. Los datos que son transmitidos desde un nodo raíz hasta un nodo central pueden cruzar por varios puntos de deteriorando el comportamiento y desempeño de la red WSN nuevamente se evidencia que la latencia impide la entrega oportuna de datos urgentes o no urgentes debido al congestionamiento de los nodos “Sink” [3].

De acuerdo a lo anteriormente mencionado, se plantea que la problemática radica en la concentración de los datos en ciertos puntos de una red de sensores tipo malla para ambientes urbanos, en las cuales se identificaron que los datos deben llegar lo más pronto posible al nodo central o nodo receptor final de tal manera que la latencia disminuya. Para este desarrollo se propone eliminar topologías robustas, y no se abordará el tema de optimización de protocolos de enrutamiento y solo se trabajará en escenarios urbanos.

Objetivos

Objetivo General

Diseñar e implementar un módulo de comunicación inalámbrico para nodos coordinadores que comunique los datos adquiridos en la red de sensores a las bases de datos en el centro de fusión de forma directa basada en una arquitectura de red con topología en estrella.

Objetivos Específicos

- Elaborar un estado del arte que contenga información científica en relación a los problemas, alternativas y soluciones de conexión entre un nodo sensor coordinador y el centro de fusión.
- Diseñar una plataforma de comunicación entre un nodo coordinador y el centro de fusión por medio de una plataforma de desarrollo abierta.
- Implementar el diseño de la plataforma propuesta, interconectando los dispositivos electrónicos utilizando: conexiones físicas, lenguajes de alto nivel, programación de microcontroladores, protocolos de comunicación inalámbrica, algoritmos de recolección de datos, servidores web locales y sistemas operativos.
- Analizar los resultados de la implementación de la plataforma, observando la respuesta y adaptación del sistema en escenarios de operación en condiciones de laboratorio.

Capítulo 1

Estado del Arte

Los avances tecnológicos han evolucionado en los últimos años, trayendo consigo nuevos dispositivos y plataformas de desarrollo para trabajar en la evolución de ciudades inteligentes. Estos avances son objeto de estudio para desarrollar nuevas aplicaciones que den soluciones en muchas áreas tales como: agricultura, espacios inteligentes, gestión de riesgo entre otros. La tecnología ha obligado que las personas estén en contacto directo con dispositivos móviles con acceso a internet, dicho contacto hace que se interactúe o estén conectados con otros elementos o cosas que lo rodean. A dicha interacción entre las personas y los dispositivos o cosas que lo rodean se llama “Internet de las Cosas” (IoT) esto sería en palabras comunes sin embargo la UIT sugiere la siguiente definición para el “IoT”: *“Infraestructura mundial al servicio de la sociedad de la información que propicia la prestación de servicios avanzados mediante la interconexión (física y virtual) de las cosas gracias al interfuncionamiento de tecnologías de la información y la comunicación”*[48]. De acuerdo a lo anterior se puede decir que las “cosas” es un término asociado a todos aquellos dispositivos electrónicos usados en la vida cotidiana para facilitar el diario vivir, los cuales entregan información de variables a los diferentes actores interesados en un proceso. En este capítulo, se explicarán los trabajos relacionados con el tratamiento de los datos en una red de sensores y los problemas dentro de ésta, en el escenario del internet de las cosas. Las redes de sensores recolectan datos de una o muchas variables análogas del medio ambiente, sin embargo, la recolección de información de estas redes debe ser guardada en forma segura facilitando el acceso.

“Una red inalámbrica de sensores (WSN) consiste de un número de nodos que combinan capacidades de medición física como temperatura o concentración de algún elemento, con capacidades de interconexión y computación. Algunas redes, como las que monitorean el medio ambiente, consisten de muchos nodos que generan datos cada segundo haciendo el total de volumen de datos generados muy grande. Sin embargo para la mayoría de aplicaciones, las mediciones de sensores individuales son de menor importancia y los usuarios están generalmente más interesados en extractos que combinan un conjunto de mediciones de datos de sensores en una estadística más simple y sólida. Es por ello que muchas organizaciones que

administran redes de sensores usan la Internet para publicar dichos extractos y de esta forma facilitar su uso” [6]. Para el transporte de información entre un nodo a otro, son utilizados los protocolos de comunicación inalámbricos los cuales deben cumplir ciertas características tales como: seguridad, bajo consumo, velocidad en la transmisión de datos entre otros.

1.1. Casos de estudio

1.1.1. Proyecto “Open WSN”

Existen muchos proyectos que trabajan el tema de redes de sensores y sus problemas pero existe uno en particular que trata el tema de protocolos y algunas recomendaciones para tratar y estudiar dicho desarrollo es el “*Open WSN*”. Este proyecto busca la implementación de una pila de protocolos de código abierto la cual se pueda implementar bajo la norma IEEE 802.15.4e “*Time Synchronized Channel Hopping standard*” y que a su vez pueda integrarse al “IoT”. Utilizando elementos de software (protocolos, simuladores) y hardware (sensores de precisión, microcontroladores, plataformas de hardware). El proyecto es desarrollado bajo el estándar IEEE 802.15.4e con conexión a internet. Para ello son utilizados los protocolos 6LoWPAN (*IPv6 over Low-Power Wireless Personal Area Network*), RPL (*Routing Protocol for Low Power and Lossy Networks*) y CoAP (*Constrained Application Protocol*) [7]. Estos protocolos trabajan sobre una red híbrida IPv4 e IPv6 tal como lo ilustra la Figura 1.1.

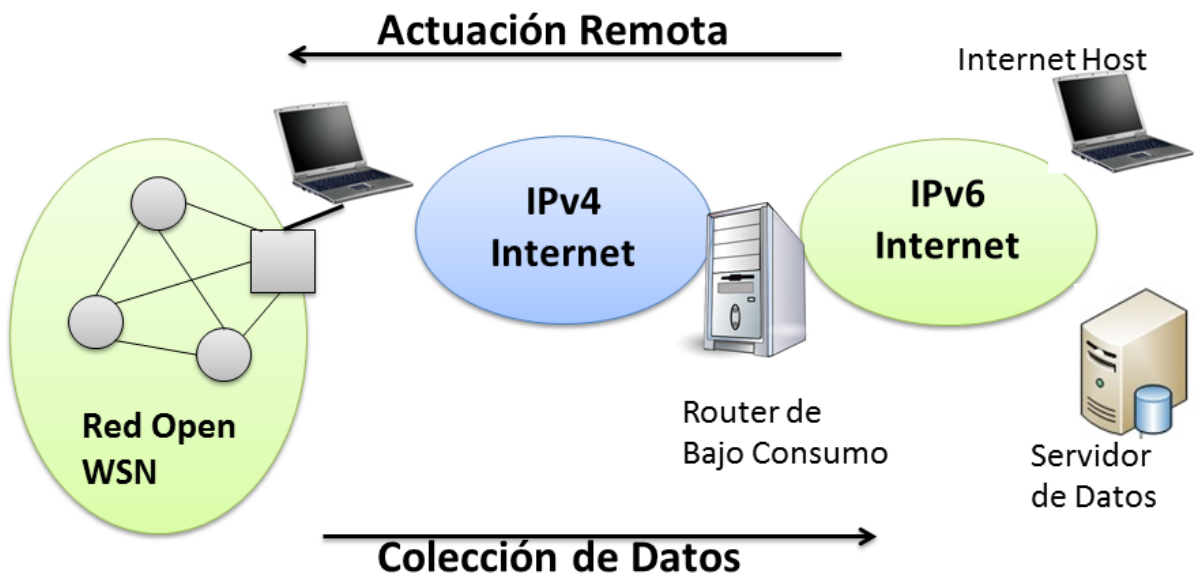


Figura 1.1: Arquitectura del proyecto Open WSN.

[7]

Para el desarrollo de este trabajo de grado se tendrán en cuenta los aspectos relacionados con los protocolos de comunicación (solo para implementación de operación mas no para análisis). El diseño del prototipo a realizar se conectará a una plataforma localizada en la nube, que para futuros proyectos se espera que éste mismo esté implementado sobre plataformas con protocolos IPv6. Por otro lado los protocolos que se recomiendan para la implementación de redes inalámbricas de sensores deben ser de bajo consumo energético asegurando la independencia de las fuentes de alimentación eléctrica.

Un protocolo de bajo consumo energético es desarrollado con el estándar IEEE 802.15.4, sin embargo, la implementación de este estándar tiene algunas desventajas las cuales van asociadas a la utilización de algoritmos de enrutamiento para la sincronización de nodos los cuales agotan la vida útil de la fuente de alimentación para este caso las baterías. Por otro lado el tema de las frecuencias y sus problemas de interferencia también inciden en forma negativa en el desempeño de la red, generalmente las redes de sensores se configuran en tipo malla lo cual hace que se evidencien estas desventajas sobre la red.

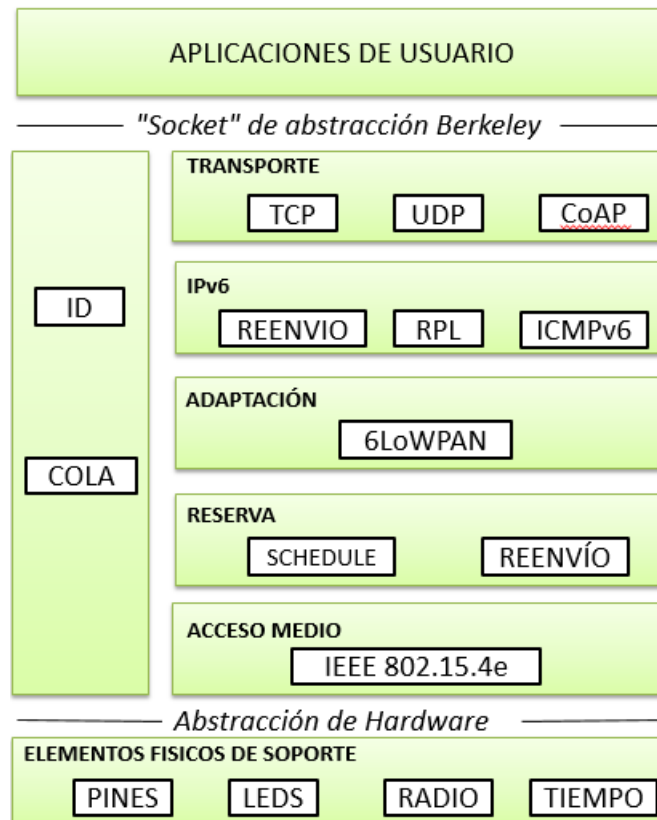


Figura 1.2: Pila de protocolos del estándar "Open WSN".

[7]

La Figura 1.2 ilustra la pila de protocolos sobre la que está implementado “*Open WSN*”. Como se mencionó anteriormente existen protocolos que requieren sincronización de nodos, los cuales agotan los recursos energéticos disponibles, sin embargo, para que este problema disminuya se pueden armar pequeñas redes de sensores no mayor a 8 nodos, que no requieran la función de sincronización en forma constante, sino que, al contrario utilice los recursos mínimos de red sin perder las condiciones óptimas y de trabajo para el adecuado funcionamiento de la red de sensores. En cuanto a la diversidad de frecuencias, los efectos de la señal se ven afectados por los fenómenos físicos asociados como a la interferencia, refracción, difracción y reflexión, entre otros.

La norma IEEE 802.4.15e busca actualizar los equipos que tengan el protocolo tradicional MAC 802.4.15 para que mejoren los problemas de sincronización de nodos. En este protocolo una trama posee ranuras que llevan información a ciertos a nodos especiales dentro de la misma red, dicha información pueden ser órdenes tales como: recibir, enviar o modo “*Sleep*”, lo cual optimiza el rendimiento de la red y mejora el consumo de energético porque cada trama lleva la información necesaria al nodo que lo requiera. Por otro lado este estándar utiliza la técnica de “*Channel Hopping*” para disminuir la interferencia generada por las otras señales del medio durante la transmisión; sin embargo según el reporte dado por “*Open WSN*” esta técnica ha tenido problemas de implementación, puesto que le hace falta adicionarle el sincronizado del canal lo cual mejoraría la comunicación efectiva entre nodos. Los protocolos utilizados en “*Open WSN*” tienen las siguientes características:

- 6LoWPAN: Cada nodo se visualiza como un host normal en internet facilitando del lado del cliente su manipulación.
- RPL: Se utiliza para mantener un enrutamiento en la malla de nodos.
- COAP: Soporta y permite la interacción “*RESTFULL*” con nodos individuales sin sobrecarga en el protocolo TCP, HTTP y UDP.

Las pruebas y resultados obtenidos, a partir de este protocolo, son óptimas puesto que en 15 (ms) (ver artículo [7]) la información que es enviada desde un microcontrolador hasta el otro nodo es rápida, sin embargo esta arquitectura suele ser muy robusta a la hora de implementar, a pesar de las ventajas mencionadas anteriormente.

1.1.2. Caso Libelium

Para el desarrollo de este numeral (1.1.2) se extrajo la información directamente de [8]. Libelium es una empresa de capital español, puntera en alta tecnología que diseña y fabrica tecnología de hardware para la implementación de redes WSN, así como, la integración de sistemas y diseño de ingeniería para llevar a cabo soluciones para ciudades inteligentes o usuarios finales en el menor tiempo posible. Una de las plataformas tecnológicas ofrecidos es el “*Waspmote*” (Tabla 1.1).

Tabla 1.1: Plataformas comerciales para desarrollo y diseño digital [8]

Productos	
	Waspote ha sido diseñado para ser de bajo consumo. Tiene conmutadores digitales permiten activar y desactivar cualquiera de las interfaces de los sensores.
	Posee sensores: CO, CO2, humedad del suelo, la presencia, la humedad, la temperatura, detección de vehículos, la radiación, la corriente, líquido, luminosidad, etc. .
	Interfaces Inalámbricas: Hay 11 diferentes interfaces inalámbricas para Waspote incluyendo larga distancia (3G / GPRS / 868 / 900MHz), de medio alcance (ZigBee / 802.15.4 / WiFi) y de corto alcance (RFID / NFC / Bluetooth 4.0).
	Protocolos Industriales: Conecte cualquier sensor en entornos industriales. Compatible con: RS-232, RS-485, Modbus, CAN Bus, 4-20 mA.
	Bibliotecas De Cifrado: Las Bibliotecas de cifrado para Waspote garantizar la autenticación, confidencialidad (privacidad) y la integridad de la información recogida por los sensores.
	6LoWPAN / plataforma de desarrollo de IPv6 (Waspote Mote Runner): Obtener conectividad IPv6 en cada nodo utilizando el código fuente pila 6LoWPAN disponible.

El mercado al que se dirige Libelium es el de Internet de la cosas (IoT). La tecnología y producto de Libelium son totalmente horizontales y transversales porque permite su integración fácil y flexible con plataformas y aplicaciones de empresas para crear soluciones para el usuario final en una multitud de aplicaciones con una misma base tecnológico [8].

“Plug & Sense!” es el nuevo waspote (Figura 1.3), que permite a los desarrolladores olvidarse de diseños electrónicos y centrarse en los servicios y aplicaciones realizando im-

plementaciones escalables teniendo costos mínimos de mantenimiento. La nueva plataforma se compone de una robusta carcasa resistente al agua y a las condiciones climáticas adversas con conexiones externas para armar los sensores, el panel solar, la antena y el cable USB con el fin de reprogramar el nodo. También, posee un módulo de seguimiento en tiempo real, que permite realizar aplicaciones de seguimiento; leer las coordenadas GPS y enviarlos mediante el uso de una petición HTTP a un servidor web, también utiliza *Googlemaps* para mostrar la ubicación en tiempo real.



Figura 1.3: Wasmote Plug&sentido!.

[8]

“*Meshlium*” (Figura 1.4) es un Linux router, el cual puede contener 5 diferentes interfaces de radio: Wifi 2,4 GHz, Wifi 5GHz, 3G/GPRS, Bluetooth y Zigbee. También, puede integrar un módulo GPS para aplicaciones móviles y vehículos y funciona con luz solar y baterías. Estas características junto con una carcasa IP67 de aluminio permite a *Meshlium* colocarse en cualquier lugar al aire libre.



Figura 1.4: Meshlium.

[8]

Libelium, se apoya en el desarrollo de proyectos importantes a nivel mundial para implementar sus productos y servicios consiguiendo experiencia en tema de redes de sensores, lo que les permite traer nuevos desarrollos, instalar nuevas tecnologías, crear nuevos dispositivos, entre otros. La tabla 1.2 describe en forma breve algunos de estos proyectos que sirven de apoyo para dimensionar el alcance del trabajo que se desarrolla en esta tesis que permite tener elementos de criterio de diseño.

Tabla 1.2: Algunos proyectos desarrollados por Libelium.

<p>En Valencia para supervisar Gestión del Ciclo del Agua.</p> 	<p>El Sistema Inteligente del Agua consiste en una red móvil inalámbrica sensor que se puede implementar rápidamente en un área en particular para vigilar la calidad del agua mediante la medición de parámetros tales como el pH, la conductividad, la reducción de la oxidación potencial y la turbidez.</p>
<p>En Serbia para el monitoreo ambiental por transporte público.</p> 	<p>Varios Waspmotes fueron instalados en los vehículos de transporte público para controlar un conjunto de parámetros ambientales a lo largo de una gran área, así como a proporcionar información adicional para el usuario final como la ubicación de los autobuses y estima tiempos de llegada a paradas de autobús.</p>
<p>En México para supervisar los motores industriales.</p> 	<p>Este sistema es capaz de controlar la temperatura y la vibración de los motores industriales, de advertencia cuando se detecta un funcionamiento anormal.</p>
<p>Agricultura Inteligente en Galicia para supervisar viñedos.</p> 	<p>Este sistema es capaz de monitorizar diferentes parámetros tales como la temperatura ambiente y la humedad, la temperatura del suelo y la humedad o humedad de la hoja.</p>

Las aplicaciones en condiciones de operación son ítems que sirven como punto de referencia para desarrollar recursos tecnológicos que faciliten el diario vivir de las personas. Por lo tanto, Libelium ofrece aplicaciones que aportan cosas muy interesantes para el tema de las ciudades inteligentes, como se describen a continuación:

Para Ciudades inteligentes:

- La salud estructural: Monitoreo de vibraciones y las condiciones materiales en edificios, puentes y monumentos históricos.

- Ruido Urbano Mapas: monitoreo de sonido en zonas de bar y zonas céntricas en tiempo real.
- Congestión de tráfico: Monitoreo de vehículos y peatones para optimizar los niveles de conducción y rutas de senderismo.
- Carreteras inteligentes: con mensajes de advertencia y desvíos de acuerdo a las condiciones climáticas y eventos inesperados como accidentes o atascos.

Para Entorno Inteligente:

- Detección de Incendios Forestales: Monitoreo de gases de combustión y las condiciones del fuego de suscripción preferente para definir las zonas de alerta.
- Contaminación del Aire: control de las emisiones de CO₂ de las fábricas, la contaminación emitida por los coches y los gases tóxicos generados en las granjas.
- Deslizamiento de tierras y la Prevención de Avalancha: Vigilancia de la humedad del suelo, las vibraciones y la densidad de la tierra para detectar patrones peligrosos en condiciones de la tierra.
- Terremoto: Detección Temprana de control Distribuido en lugares específicos de los temblores.

Para Agricultura inteligente:

- Vino de Calidad: Mejora de monitoreo de humedad del suelo y el diámetro del tronco en los viñedos para controlar la cantidad de azúcar en las uvas y la salud de la vid.
- Casas Verdes: Control de las condiciones micro-climáticas para maximizar la producción de frutas y hortalizas y su calidad.
- Estación Meteorológica de la Red: Estudios de las condiciones meteorológicas en los campos para el pronóstico.

1.1.3. Otros Casos de Estudios

Los proyectos desarrollados alrededor de la gestión de riesgo en diferentes países, tienen un impacto a nivel tecnológico [9], por ello, las redes inalámbricas de sensores han sido utilizadas, como es el caso de India en donde los sistemas de alerta temprana para avalanchas de lodo [10] son instalados sensores de profundidad en un esquema híbrido. Las WSN también se emplean para el estudio de inundaciones en los ríos de Brasil, donde investigadores

emplearon una red basada en módulos Xbee¹ instalando los sensores en las riberas comunicándose con una estación central [11]. Las redes de sensores están presentes en zonas de alto riesgo de inundación por precipitaciones y también en redes de observación hidrometeorológicas para sistema de alerta temprana de crecidas repentinas.

Otro trabajo similar se realizó en la escuela Politécnica del Ejército en Ecuador en donde se analizó el comportamiento y tiempos de respuestas de una red de sensores aplicadas a monitoreo de volcanes analizando el desempeño de la red, arrojando los siguientes resultados: según este estudio se determinó que la topología por teselación triangular es una de las que presenta mejores características para este tipo de aplicaciones, por otro lado el software que mejores condiciones presentó para este estudio fue el ns-2 (*Network Simulator*)² en cuanto al hardware que mejor se adaptó a la prueba fue el Crossbow de la empresa *Memsic* el cual brinda las mejores prestaciones y características los cuales permiten aplicarlos en monitorización volcánica en tiempo real [12]” para mayor información en relación a este trabajo remitirse a la cita bibliográfica.

En cuanto al contexto nacional, en las selvas del Chocó Colombiano, se realizó un estudio donde se analizó el comportamiento de las fuentes de alimentación o baterías de una red de sensores que monitoreaban árboles en un área específica del territorio selvático. Los resultados arrojaron que la batería pudo durar en teoría hasta un año, sin embargo el comportamiento de duración de batería en los “nodos router³” fue deficiente, las simulaciones y análisis del diseño sugirieron que la carga solo duraría un lapso de 21 días, incluso, aumentando la capacidad de la batería a 2500mA/h. El problema radicó en la forma de operación de los “nodos router”, por lo cual puede que existan dos posibles soluciones: se mejora la fuente de energía, o se mejora el esquema de operación de la red para minimizar el gasto energético”[13] .

En la gestión de riesgo ambiental se cuenta con datos provenientes de redes internacionales las cuales están disponibles en tiempo real a través de internet o de enlaces satelitales. Para la transmisión y obtención de esta información se utiliza: la telefonía radial, radio “UHF/VHF” o “PRD GOES”. No obstante los principales factores para determinar qué tipos de comunicación es la mejor para la transmisión de los datos son: la velocidad de transferencia de datos, disponibilidad de energía eléctrica, garantía de transmisión de datos, ubicación y disponibilidad de infraestructura de telecomunicaciones [14].

¹Dispositivos utilizados en redes de sensores.

²ns es un simulador de redes basado en eventos discretos. Se usa principalmente en ambientes educativos y de investigación. Permite simular tanto protocolos unicast como multicast y se utiliza intensamente en la investigación de redes móviles ad-hoc. Implementa una amplia gama de protocolos tanto de redes cableadas como de redes inalámbricas. La versión actual, ns-3, está diseñada para soportar todo el flujo de trabajo de la simulación desde la configuración hasta la recolección y análisis de tramas.

³Una red Xbee es utilizada en redes inalámbricas de sensores en donde el “nodo router” está asociado a un nodo coordinador por lo tanto en una topología en árbol el nodo router es el encargado de recibir y posteriormente enviar información al nodo coordinador o en su defecto a otro nodo router.

1.1.4. Proyectos relacionados con la problemática identificada.

Korea es uno de los países que designa mayor inversión económica en la investigación de redes inalámbricas de sensores, lo cual refleja la importancia del estudio de esta temática tanto para un sistema de software como de hardware, debido a que juegan un papel indispensable dentro de las ciudades inteligentes.

Asimismo, un estudio de la Universidad de Bologna establece una propuesta de solución a una problemática existente en las WSN, centrada en el consumo de energía y la recolección masiva de datos, dicho planteamiento busca la forma de integrar una plataforma MANET⁴ con las WSN, éste arroja conclusiones interesantes, entre las cuales se destacan la optimización de entrega de datos urgentes a estaciones centrales de administración permitiendo una entrega segura, rápida y confiable de los datos urgentes, el proyecto habla de una superposición de redes MANET sobre redes de sensores inalámbricas. Los Nodos MANET deben dinámicamente, auto-organizarse y explotar oportunamente a sus vecinos actuales para proporcionar la recopilación de datos rápidamente, incluso en condiciones difíciles de red. En segundo lugar, el protocolo de organización MANET debe ser localizado vía web, sin embargo, las MANETs sufren de graves problemas de robustez y de ancho de banda a medida que aumenta la longitud de ruta de punta a punta, los nodos MANET deben organizarse en grupos locales, sin necesidad de interacción de larga distancia. En tercer lugar, debe ser tolerante a la movilidad del nodo” [3].

Por otra parte, las redes de sensores implican problemas de latencia en la entrega de los datos por lo tanto, una red a escala debe ser capaz de soportar las exigencias que en un momento determinado se requiera, como transmitir información urgente que relacione el desbordamiento de un río en zona de alto riesgo.

De igual modo, la complejidad de las tecnologías y la rigidez de las redes limitan las estrategias de integración de dispositivos pero tal situación pasa desapercibida a los usuarios, en tanto que las redes de sensores hacen parte de la alta gama que ofrecen hoy las tecnologías de la información, haciendo posible la integración de sistemas, es decir, el “Internet de las cosas (IOT)”, lo cual permite la interoperabilidad de tecnologías reconociendo comunicación sin limitaciones y heterogeneidad de extremo a extremo. La importancia de los sensores se ve reflejada en un trabajo realizado en la India, donde la empresa IBM invirtió para un proyecto de investigación el cual buscaba monitorear las lluvias y pronosticar inundaciones, para ello recurrió a una red de sensores de bajo costo. Las redes de sensores heterogéneas pueden llegar a ser homogéneas [4] de acuerdo a su implementación y diseño, no obstante, la problemática de escalabilidad es persistente.

⁴Las Mobile ad hoc network (denominadas también MANET como acrónimo de Mobile ad hoc network), se trata de una red de dispositivos conectados por wireless y que poseen propiedades de auto-configuración, además de poseer cierta movilidad.

Adicionalmente, existen otros factores que se deben tener en cuenta como la limitación de la conexión mediante cortafuegos por parte de algunos administradores de la red, además las direcciones son provenientes de un NAT⁵, es importante percatarse que todo este tipo de aseveraciones se hacen a partir de un escenario grande como una ciudad Ubicua.

El estudio de las redes de sensores ha cobrado importancia en los últimos años, convirtiéndose en el eje central de las ciudades inteligentes, pero vale preguntar ¿qué elementos o dispositivos son utilizados para conformarlas?. El protocolo Zigbee de comunicación inalámbrica es uno de los más utilizados debido a su fácil manejo y a sus propiedades de bajo consumo de energía. Asimismo, en el “*Beijing Institute of Technology*”, se realizó un estudio en el que presentó un prototipo de conexión de estos dispositivos para ser utilizados, controlados y manipulados a través de una página Web. El sistema se dividió en tres capas, en primer lugar, la capa de percepción (sensores), la capa de servidor web (Bases de datos y servidor basado en PHP) y la capa de aplicación (dispositivos terminales con navegador). Algunos datos interesantes arrojados por este estudio son las pruebas de escalabilidad, pues en este sistema no solo se pueden conectar más sensores en diversas formas sino que también pueden aplicarse a otras áreas de recolección de datos y procesos de supervisión debido a las técnicas, principios y tecnologías claves con un alto grado de versatilidad.

Finalmente, para la toma de decisiones en las funciones de emergencia, el sistema da prioridad al procesamiento de ésta [15], no obstante, aún faltan elementos por mejorar en este estudio respecto a la escalabilidad, ya que los sensores no fueron expuestos a grandes cantidades de datos, sino que se hicieron pruebas puntuales; entonces vale anotar dentro de las recomendaciones, la inclusión de servicios más autónomos y que ofrezcan muchas más posibilidades en sus aplicaciones. Este estudio permitió establecer tres condiciones necesarias: sensores, aplicación, servidor y protocolo de comunicación.

1.2. Protocolos de comunicación para WSN.

IEEE ha creado varios estándares dentro de la familia de 802.x que se han adoptado para las comunicaciones de datos inalámbricas. Hay muchas aplicaciones que requieren establecer una red de bajo alcance con bajas tasas de transmisión. Estas redes se denominan LR-WPAN. (*Low Rate Wireless Personal Area Network*), el IEEE802.15.4 es un estándar para las LR-WPAN que provee una soluciones simples y de bajo costo para el manejo de redes de sensores y activación de actuadores, así como, aplicaciones en domótica, monitoreo, industria, hospitales y hoteles [26]. El protocolo fue definido para estandarizar redes “WSN”.

⁵NAT es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles. Consiste en convertir, en tiempo real, las direcciones utilizadas en los paquetes transportados.

1.2.1. Protocolo Zigbee y la “Alliance Zigbee”

En la actualidad la interconectividad entre los dispositivos electrónicos toma más fuerza, por lo cual la industria se ve en la necesidad de crear de redes de comunicación para hogares, hospitales, escuelas, instituciones entre otros. Las tendencias indican que en un futuro cercano el “IoT” abarcará gran parte de los procesos en el mundo, en donde todos los dispositivos electrónicos estarán conectados entre si. Por ejemplo, hoy en día hay sensores instalados previamente dentro del cuerpo de una persona para medir su niveles de glucosa, ritmo cardíaco, temperatura y enviar los datos a un centro de fusión o alertar en una red por una situación de emergencia médica. Por ello entre otras razones, las grandes empresas a nivel mundial desarrolladoras de dispositivos de instrumentación y medida crearon la “ALLIANCE”, encargada de generar un estándar de comunicación; sino que se puedan generar códigos abiertos de comunicación en los cuales se pueden hacer diseños o arreglos de hardware sin ningún inconveniente y sin pensar en el tema de las licencias. La innovación y creatividad dentro de los productos de la alliance trae beneficios como interoperabilidad de dispositivos estandarizados, actualización de los productos existentes o añadir nuevas características innovadoras



Figura 1.5: Productos de la Alliance.

El estándar Zigbee se basa en la norma IEEE 802.15.4, la cual opera en bandas sin licencia en todo el mundo a 2,4 GHz (global), 915Mhz (americanas) y 868Mhz (Europa). Este ofrece varias velocidades de datos como es el caso de 250Kbps a 2,4 GHz (16 canales), 40Kbps a 915Mhz (10 canales) y 20Kbps a 868Mhz (1 canal). Las distancias de transmisión son amplias y van desde 10 a 1600 metros la cual depende de la potencia y las condiciones ambientales, otras características del estándar Zigbee son el bajo consumo energético que permita usar equipos a batería, instalación de redes de sensores en edificios, bajo costo de dispositivos e instalación y mantenimiento de ellos.

Existen muchos estándares que se pueden usar en redes de corto alcance o personales [Figura 1.6] tales como el 802.11 y Bluetooth. Cada uno de estos está desarrollado para una clase de aplicación determinada orientadas a redes de área personal. Zigbee es el estándar más usado para redes de sensores y actuadores que deban operar a batería. Las redes de área

personal, están pensadas para proveer una comunicación en un espacio personal de hasta 10 m de radio sin necesidad de infraestructuras robustas. Las redes de área personal se dividen en redes de alta, media y baja velocidad, IEEE802.15.3 es un ejemplo de red de alta velocidad que puede ser usada para transmitir video desde una cámara a un TV cercano.

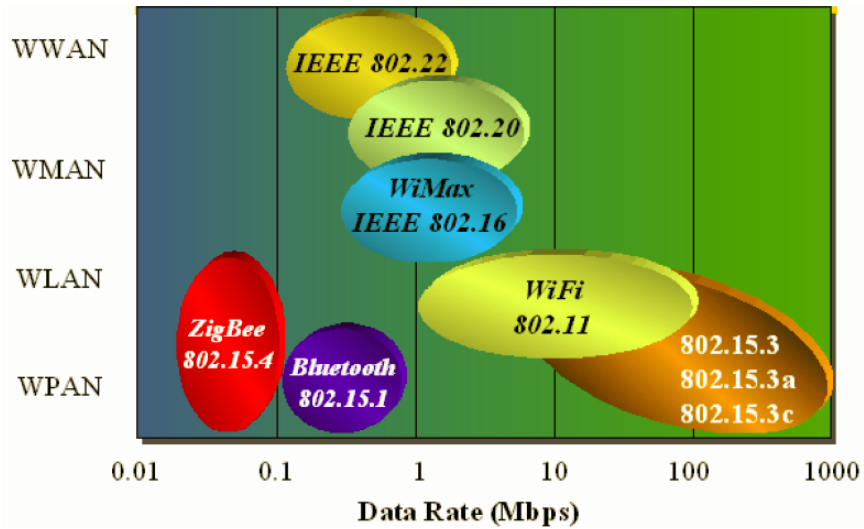


Figura 1.6: Clasificación de las redes Inalámbricas. [26]

La Figura 1.7 ilustra y diferencia los protocolos Zigbee y Wifi y sus componentes estandarizados lo cual permite distinguir entre estas dos tecnologías de comunicación inalámbricas.

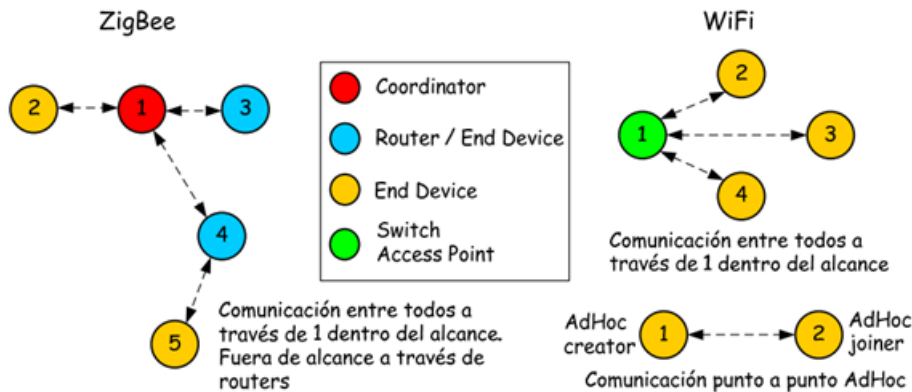


Figura 1.7: Componentes de una red WiFi y Zigbee. [16]

Bluetooth es un ejemplo de estándar de velocidad media. Puede ser usado para transmisión de música de alta calidad desde un equipo de audio a auriculares inalámbricos. También se emplea para conectar periféricos a computadoras. ZigBee, con una velocidad de datos máxima de 250 kbps, es considerada una red personal inalámbrica de baja velocidad.

1.2.2. Módulos de comunicación Inalámbricos de Digi.

Digi internacional es una compañía que ofrece soluciones para la conectividad serial, sus productos están divididos en familias: conexión drop in, embebidos, conectividad inalámbrica, zero clients, servidores de consola, servidores serie, Video/sensores, conectividad serie y USB. La empresa Digi tiene en el mercado comercial de dispositivos electrónicos inalámbricos de comunicación o tres elementos de hardware con características similares pero con diferentes protocolos de comunicación [Tabla 1.3]. De igual forma en el reporte técnico de la tesis se detallan algunos aspectos en relación a los módulos Xbee serie 2 de Digi internacional en el capítulo “*Descripción de los módulos Xbee de Digi*”

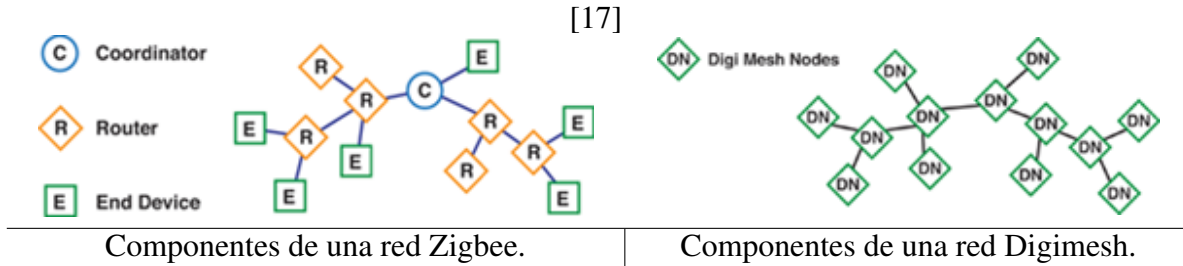
Tabla 1.3: La serie Xbee de Digi internacional.

Módulos Xbee De Digi	
Modulo	Protocolo
Xbee Serie 1	802.4.15
Xbee Serie 2	Zigbee
Xbee Serie 3	Digimesh

1.2.3. Esquema de un módulo Xbee y una red Zigbee.

Una red Xbee puede ser dos clases: Zigbee y Digimesh. Para el caso de la red Zigbee siempre debe haber un coordinador quien inicia la red y a él se conectan router's y end device, para el caso de la red Digimesh todos los nodos están en igualdad de condiciones y todos se ven con todos pero la mayoría de los módulos Xbee tienen en común las mismas características en sus pines. La Tabla 1.4 compara los elementos entre una red Digi y una red Zigbee en este trabajo se hace necesario contar con elementos de plataformas abiertas, por lo tanto los se escogen elementos de red Zigbee.

Tabla 1.4: Comparación entre los elementos de una red Zigbee y una red Digi.



1.3. Ordenadores de Placa Reducida.

Los ordenadores de placa reducida son elementos que poseen las mismas características de un ordenador de escritorio y pueden servir en una aplicación embebida. En los años 80 apareció el concepto de “*System on Chip*” (SoC), el cual es un integrado que poseía todas la características de un computador sin embargo por costos no pudo ser implementado comercialmente. A partir, del año 2008 empezaron a aparecer las plataformas “*open hardware*” para diseño electrónico digital pero se limitan solo a lo que la placa pueda realizar. Las plataformas “*open hardware*” fueron creadas para facilitar el diseño de aplicaciones que incluyan algún tipo de control electrónico sobre algún dispositivo, proceso o sistema. Sin embargo a este tipo de dispositivos, les hacía falta un componente de software en el cual se pudiera trabajar algo más que un diseño de hardware, por esta razón aparecen los ordenadores de placa reducida contienen sistema operativo y puertos “*In/Out*”, lo cual hace posible que un sistema pueda ser escalable ofreciendo las disposiciones de lenguajes de alto nivel para propósitos mucho más complejos y a escala. Los ordenadores de placa reducida son una computadora completa en un sólo circuito, el diseño se centra en un sólo microprocesador con la RAM, E/S y todas las demás características de un computador funcional en una sola tarjeta que suele ser de tamaño reducido, y que tiene todo lo que necesita en la placa base. En el reporte técnico de la tesis adjunto a este documento se describen las especificaciones técnicas de los ordenadores de placa reducida encontrados comercialmente hasta la fecha (Capítulo “*Descripción de ordenadores de placa única*”).

1.4. Aspectos del Problema

De acuerdo con los casos expuestos anteriormente, se analiza que la implementación de las redes inalámbricas de sensores tienen muchos aspectos de implementación. En la tabla 1.5 se describen algunos criterios de diseño:

Tabla 1.5: Aspectos o criterios de Diseño.

Aspecto	Descripción
Confiabilidad	Los datos deben ser enviados y tener la certeza de que llegaron a través de un checksum.
Seguridad	La trama o símbolo enviado debe tener un número de verificación esto con el fin de que no pueda ser alterada o desviada de su destino.
Escalabilidad	El sistema debe estar ajustado a un número de nodos que respondan ante la necesidad de un área geográfica establecida.
Latencia	Los datos no deben demorar mucho tiempo en el canal de transmisión puesto que el delta de (t) producido por esta puede ser alto, haciendo crecer la cola de datos en forma exponencial.
Desvanecimiento	Debido al efecto multitrayectoria para ambientes urbanos, la señal que contiene la información se ve afectada. Por lo tanto el diseño debe asegurar que en un radio de x metros la información llegue.
Compatibilidad	Los dispositivos utilizados en una red de sensores deben estar trabajando bajo los mismos parámetros de operación.
Consumo	Los protocolos y dispositivos deben garantizar la durabilidad de la vida útil de una batería o fuente independiente.
Sincronización	Los datos deben llegar en tiempos adecuados

El trabajo de tesis pretende generar alternativas que aporten a la solución del “planteamiento del problema” a través de las herramientas que mejor se ajusten al medio, las técnicas que mejor se adapten, el diseño apropiado y la tecnología que esté al alcance de la mano cabe anotar que estos aspectos contienen de fondo muchas dificultades de tipo técnico, tecnológico, diseño, implementación, seguridad, condiciones de operación entre otros. En este primer capítulo se estudiaron los aspectos relacionados con las plataformas tecnológicas para el diseño e implementación del dispositivo final. Para efectos prácticos de este trabajo, se diseñará, implementará y analizará un dispositivo que capture los datos de una red de sensores y los cargue en forma directa en un servidor en internet, proponiendo y diseñando un sistema distribuido de nodos coordinadores. Este sistema no solo optimizaría el flujo de datos sino que sería escalable y eliminaría parte del retardo en la entrega de datos. La solución en este

trabajo de tesis es presentar alternativas de implementación y análisis técnico que disminuya la latencia y aumente la escalabilidad en una red inalámbrica de sensores a través de una arquitectura de red compuesta por elementos de bajo costo comercialmente encontrados en el mercado.

Capítulo 2

Diseño

De acuerdo con el capítulo 1 “Estado del Arte”, se puede observar que para el desarrollo del proyecto existen muchas alternativas tecnológicas. Por ello, de acuerdo a estos recursos tecnológicos se diseñará un dispositivo que capture los datos de una red de sensores y los suba en forma directa a la nube. Se emplearán elementos de hardware y software los cuales se irán describiendo en este capítulo. La propuesta de hardware se divide en tres elementos o bloques que representan el diseño final, mientras que la propuesta de software es representada en dos diagramas de caso de uso y un modelo entidad relación para la base de datos. Para el caso del diseño de hardware se escogen elementos de bajo costo puesto que para “internet de las cosas” es necesario que el factor económico no sea muy elevado tal que se puedan hacer sistemas distribuidos de cosas interconectadas entre si enviando información de variables, sin embargo, los elementos también deben poseer características óptimas de operación en que los sistemas no pierdan sus condiciones naturales de implementación teniendo en cuenta el tema de escalabilidad, para ello las opciones dadas por los ordenadores de placa única son un alternativa importante para el desarrollo del prototipo. Los bloques finales del diseño están descritos en la Tabla 2.1 y su distribución se encuentra asociada a los elementos de hardware:

Tabla 2.1: Diseño del prototipo por bloques.

Bloque	Elementos de Hardware
Bloque 1 (Nodo Sensor) Transmisor	Microcontrolador PIC1937, Módulo Xbee serie dos configurado como Router, y Sensor
Bloque 2 (Nodo Coordinador) Receptor	Raspberry Pi modelo B y Módulo Xbee serie dos configurado como Coordinador
Bloque 3 (Centro de Fusión) Base de Datos	Servidor contratado en la nube

Como se puede observar estos bloques contienen elementos de hardware que interconectados constituyen el diseño del prototipo, pero a parte de estos elementos aparecen otros llamados elementos software los cuales se estudiarán en el capítulo 4. Por otro lado, los dispositivos de comunicación inalámbrica escogidos son plataformas que poseen protocolos abiertos y de bajo consumo al igual que los elementos de o dispositivos programables. El diseño de software es a través de plataformas de código abierto de tal manera que si se desea realizar una implementación en ambientes aplicados, no haya necesidad de preocuparse del tema del licenciamiento, también el diseño posee algoritmos con altos niveles de abstracción de tal manera que sea entendible y claro. Inicialmente se describirán los elementos escogidos para el diseño.

2.1. Ordenador “Raspberry Pi model B”

Como se estudió en el capítulo, anterior los ordenadores de placa reducida son los ideales para este tipo de aplicaciones de redes de sensores por su bajo costo, bajo consumo de energía, sistema operativo, pines físicos de conexión y la posibilidad de interactuar con lenguajes de alto nivel entre los cuales se destacan App Inventor, Live Code, Java, JavaScript, Python (Recomendado), Ruby, HTML 5, etc. El criterio para escoger la Raspberry pi modelo fue el bajo costo (65\$ US) respecto a las otras plataformas encontradas comercialmente (Ver reporte Técnico de tesis “Descripción de ordenadores de Placa Única”). Cuando se habla de Internet de las cosas generalmente se traduce realizar la interconexión de muchos dispositivos, para avanzar en este sentido los dispositivos interconectados deben ser de fácil acceso económico.



Figura 2.1: Raspberry Pi Modelo B.

Los pines físicos de la Raspberry son conocidos como “*General Purpose Input/Output*” GPIO y el sistema operativo recomendado para instalar en este dispositivo es el Raspbian derivado de la distribución Debian Wheezy (Núcleo o Kernel de Linux). La Figura 2.2 muestra

los pines físicos de la Raspberry modelo B [Figura 2.1], de los cuales el pin 14 – 15 son los habilitados para realizar comunicación serial asíncrona.

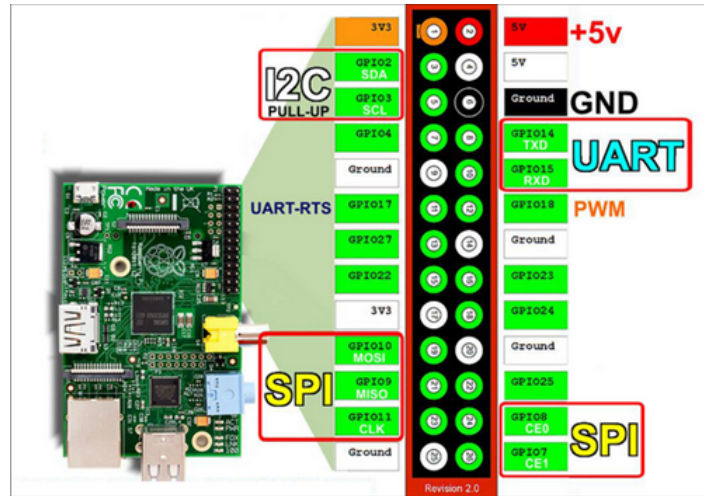


Figura 2.2: Esquema de pines de la Raspberry.
[19]

No obstante, existen otros aspectos a tener en cuenta entre los cuales se destacan las ventajas y desventajas al utilizar la Raspberry para este caso el factor más importante a tener en cuenta es desarrollar un prototipo de bajo costo pero que ofrezca todos los elementos tecnológicos para desarrollar este trabajo de tesis [Tabla 2.2].

Tabla 2.2: Ventajas y desventajas de utilizar Raspberry Pi modelo B.

Raspberry Pi modelo B	
Pro	Contra
<p>Bajo Costo (65\$US)</p> <p>Procesador ARM de 700 MHz</p> <p>Sistema Operativo</p> <p>Pines GPIO</p> <p>Consumo de Energía (5W max)</p> <p>Interfaz Ethernet</p> <p>Existe documentación</p> <p>Pines TX y RX para UART</p>	<p>Algunas aplicaciones de alto nivel no son compatibles con este dispositivo.</p>

La Raspberry ofrece la posibilidad de un sistema escalable debido a sus propiedades de hardware y software en donde se puede implementar una red distribuida de nodos sensores

capturando datos en forma simultánea o progresiva, a diferencia de las plataformas que no poseen sistema operativo sino que dependen de otros elementos para su funcionamiento como ejemplo existe “Arduino” el cual tiene dos limitantes la primera es un sistema que depende solo de él y de los que su programa almacene en memoria ROM y la segunda tiene que ver con el único lenguaje de programación que tiene (Lenguaje “C”). A pesar de que la Raspberry es la opción escogida para desarrollar este prototipo, no es la única en el mercado existen muchos otros ordenadores de placa única que ofrecen iguales o mejores características técnicas para condiciones de implementación.

2.2. Módulos inalámbricos con protocolo Zigbee.

Como se pudo apreciar en el capítulo 1, los módulos de comunicación inalámbrica que más se ajustan a las necesidades son los módulos Xbee serie 2 fabricados por Digi International porque poseen protocolo Zigbee el cual es un estándar creado para comunicaciones inalámbricas de sensores, industriales, espacios inteligentes, etc. Además es un protocolo abierto y compatible con muchos dispositivos comercialmente hablando, además es de los dispositivos más económicos comercialmente hablando que se encuentra en el mercado. La Figura 2.3 detalla la conexión eléctrica de pines de los Módulos Xbee serie 2. Para el caso de la aplicación del prototipo es necesario utilizar los pines TX y RX.

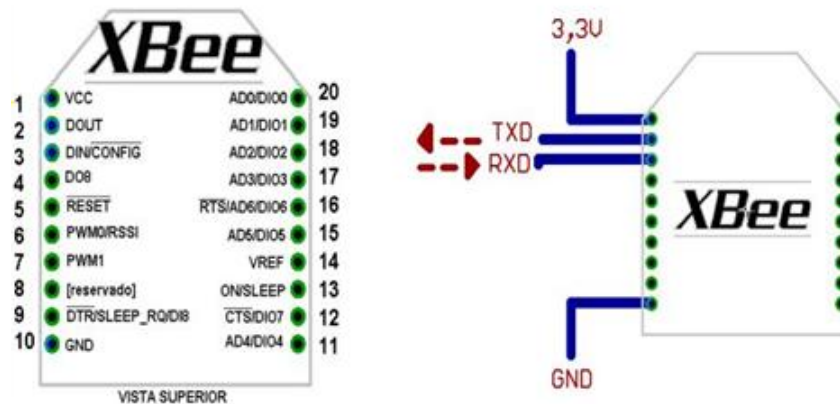


Figura 2.3: Diagrama de pines de un Módulo de Xbee serie 2.
[20]

El módulo Xbee de acuerdo al diseño debe comunicarse con el microcontrolador a través de sus pines TX y RX tal como se describe en la Figura 2.4 a través de un par cruzado TX-RX—RX-TX sin embargo los módulos Xbee son configurables a través de un software especializado el cual permite su programación para que estos módulos operen en dos formas: Modo comandos AT y Modo comandos API.

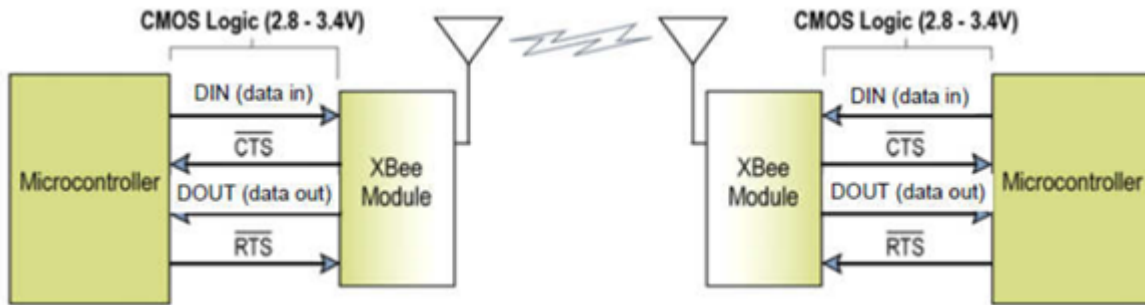


Figura 2.4: Conexión Xbee – Microcontrolador.
[21]

Para el caso de los comandos AT la Tabla 2.3 describe los pro y contras al utilizar comandos AT.

Tabla 2.3: Ventajas y desventajas de utilizar comandos AT.

Comandos AT	
Pro	Contra
Simple Transparente, se recibe lo que se envía	Se configura usando comandos AT Se debe entrar en modo comando para cambiar destino con un comando AT No se indica en la recepción, el origen Modo comando es lento y tedioso

Trabajar con comandos AT es muy sencillo, pero no ofrece las posibilidades de hacer un sistema a escala debido a sus limitaciones, para ello las tramas API ofrecen más posibilidades y opciones de configuración que se ajustan a las necesidades de implementación [Tabla 2.4].

Tabla 2.4: Ventajas y desventajas de utilizar tramas API.

Tramas API	
Pro	Contra
Simple cambio de destino Indicación de origen y destino de un dato Datos pueden contener comandos AT Permite configurar remoto Permite monitorear remoto.	Complejas de armar

Las tramas API detallan mucho más la información que se tramite y recibe, adicionalmente se pueden realizar tareas más especializadas tales como acceso remoto, monitorización automática de variables, chequeos de estados de una entrada y envío de información carácter a carácter, la Figura 2.5 ilustra la distribución de una trama API.

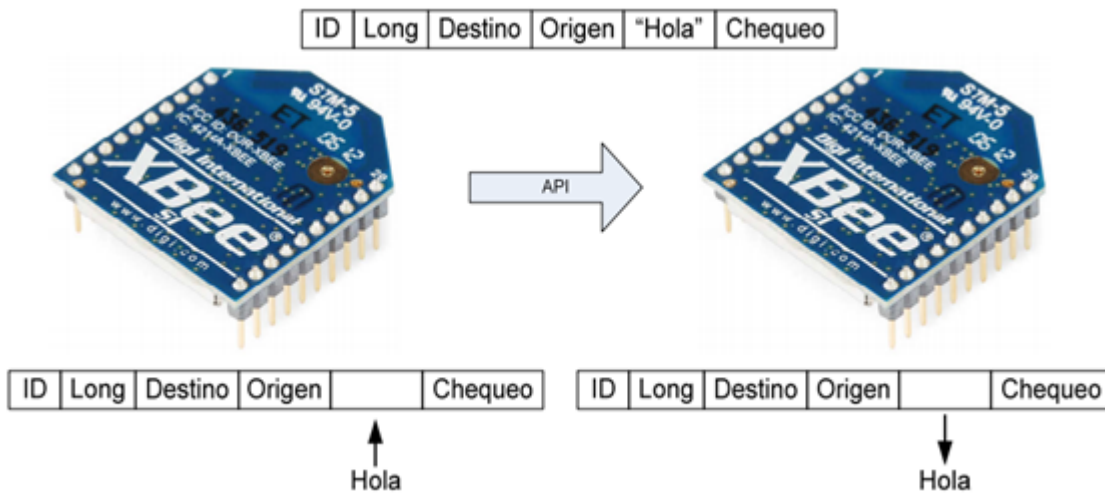


Figura 2.5: Esquema general de una trama API. [16]

De la información de las tramas API la más importante es la dirección origen de donde proviene la trama y la información del sensor, estos datos se constituyen en el eje central de la base de datos que se encuentra en el nodo coordinador o receptor.

Los módulos Xbee por sus características de fabricación se ajustan a las necesidades de la aplicación para ello la Tabla 2.5 describe algunas ventajas y desventajas al utilizar estos módulos.

Tabla 2.5: Ventajas y desventajas de utilizar los Módulos Xbee serie 2.

Módulos Xbee Serie 2	
Pro	Contra
Bajo costo (25 \$US) Alcance 300 Metros línea de vista Bajo consumo energético Protocolo Zigbee Comunicación a través de comandos API Comunicación a nivel de MAC	Velocidad lenta de transmisión en relación a otros dispositivos 250KBps.

2.3. Descripción de los microcontroladores PIC

Un microcontrolador es un circuito integrado que contiene en el mismo encapsulado las tres unidades funcionales básicas de un ordenador, a saber, unidad central de procesamiento (CPU), memoria y periféricos de entrada/salida [Figura 2.6]. Se fabrican para reducir el costo económico y de consumo de energía de un sistema particular, acotando las características del mismo a las necesidades del sistema donde se va a utilizar. El microcontrolador incide de manera significativa en el diseño del dispositivo porque posee un convertor análogo-digital que recibe la información del sensor, un módulo de comunicación serial USART para enviar la información de las variables al dispositivo inalámbrico como ventaja se destaca que al usar el microcontrolador este solo se dedica a dos tareas recibir y enviar la información al módulo transmisor, lo cual optimiza el proceso.

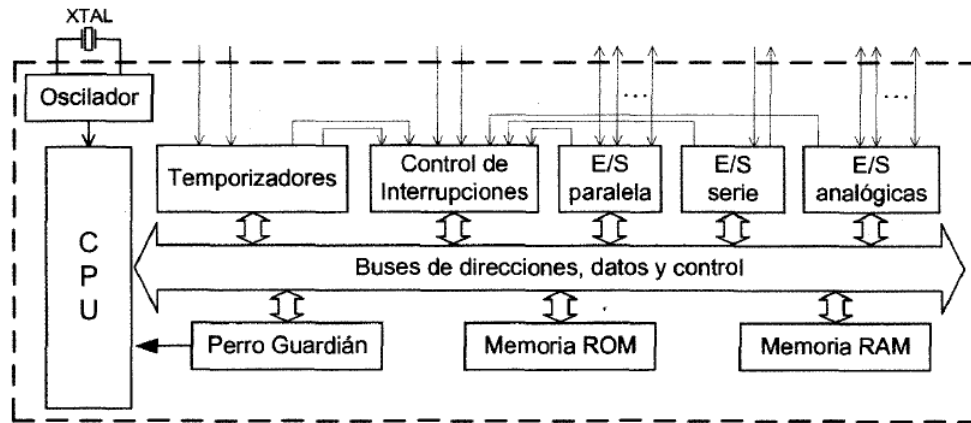


Figura 2.6: Esquema básico de un microcontrolador.

[22]

Para este trabajo, el fabricante de microcontrolador que se usará es “*microchip*” utilizando la familia 16F perteneciente a la de gama media. Pese que en la actualidad existen multitud de fabricantes de microcontroladores en el mercado, en el ámbito educativo y aficionado los microcontroladores más utilizados son los PIC de la empresa Microchip y los AVR de Atmel. Los microcontroladores PIC son una familia de microcontroladores RISC fabricados por Microchip basados en el PIC1650 desarrollado originalmente por General Instruments. Las características son precisadas a continuación:

- Arquitectura Harvard (el área de código y datos están separadas).
- Un número de instrucciones reducido y de longitud fija.
- Instrucciones se ejecutan en un ciclo de máquina (ciclo extra para saltos).
- Todas las posiciones de la RAM funcionan como registros de origen y/o de destino.
- Una pila de hardware para almacenar instrucciones de regreso de funciones.

-El contador de programa está también relacionado dentro del espacio de datos.

Para el desarrollo del dispositivo se adoptó el microcontrolador PIC16F1937 [Figura 2.7] el cual posee los siguientes pines de conexión eléctrica.

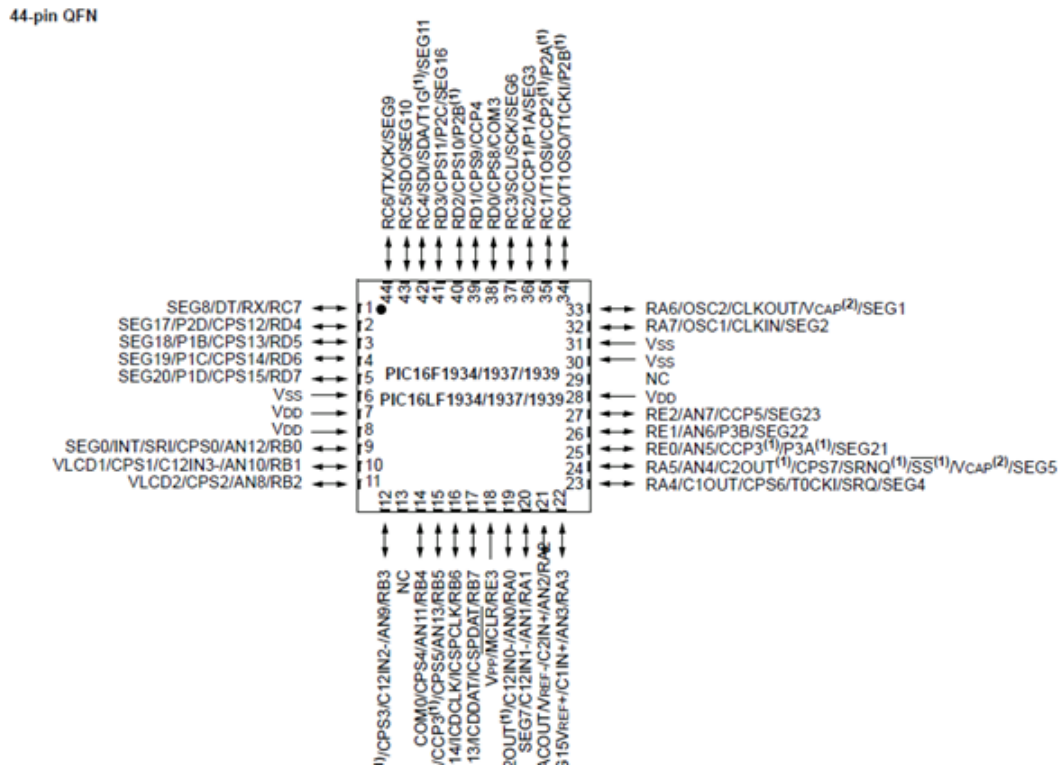


Figura 2.7: Diagrama de pines del PIC16F1837 encapsulado SIP. [23]

De acuerdo a los puertos que se utilizaran en el diseño, éstos serán los pines de TX y RX para la transmisión y recepción de datos y un canal análogo para recibir los datos procedentes del sensor [Tabla 2.6].

Tabla 2.6: Pines del PIC16F1937 utilizados en diseño.

PINES PIC16F1937		
Pin	Nombre	Descripción
44	TX	Transmisión de datos
1	RX	Recepción de datos
19	ANSELA1	Canal Análogo 0

Para poder programar un microcontrolador de microchip se necesita el programador especializado de microchip “Pickit 3” el cual debe ser conectado a unos pines específicos del microcontrolador [Figura 2.8] pero, puede utilizarse cualquier ambiente de desarrollo integrado que sea compatible con el este dispositivo programador.

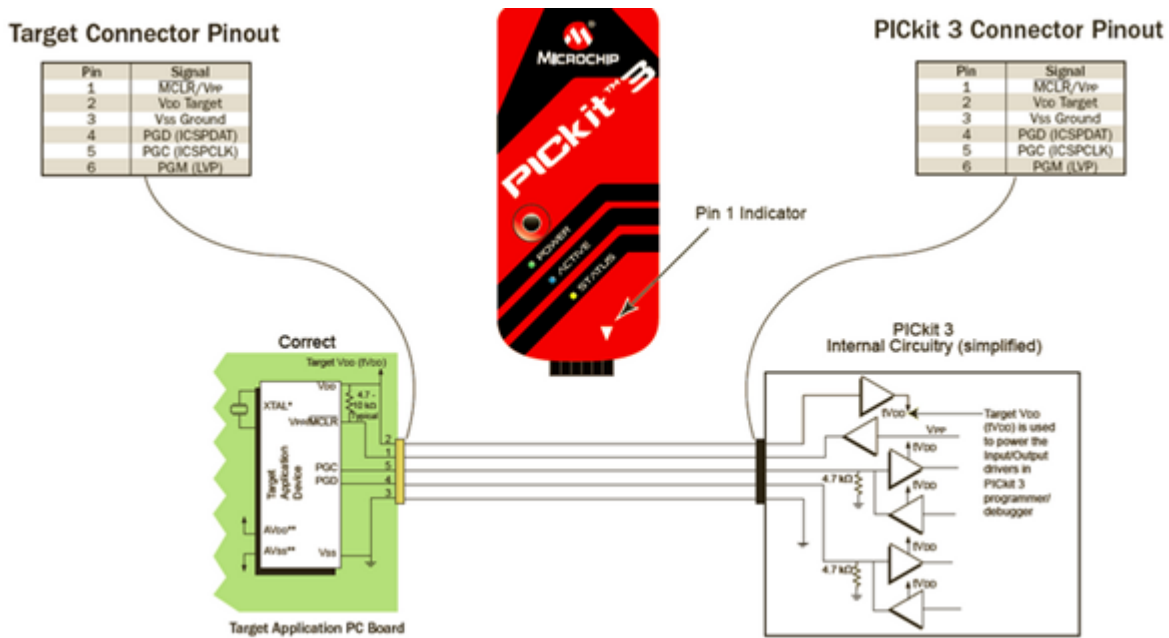


Figura 2.8: Bloque físico para programar un Microcontrolador. [24]

La única desventaja que posee hasta ahora este dispositivo microcontrolador es la poca documentación existente por lo cual el diseño de programación debe hacerse desde cero [Tabla 2.7]

Tabla 2.7: Ventajas y desventajas de utilizar el microcontrolador PIC16F1937.

Microcontrolador PIC16F1937	
Pro	Contra
Bajo Costo (3\$US)	Algoritmos de programación complejos
Velocidad de procesamiento 16 MHz	Existe poca documentación.
14 canales de conversión A/D	
Encapsulado DIP y SIP	
Módulo ESUART (TX y RX)	

Este dispositivo posee enormes ventajas porque, a pesar de ser de gama media, posee

características de un microcontrolador de gama alta y es ideal para recibir las señales de uno o varios sensores a través de sus pines de conversión analógicos-digitales y transmitir las a través de su módulo EUSART de transmisión y recepción de datos asíncrona.

2.4. Diseño del prototipo de Hardware y Software

La propuesta de diseño está constituida por tres bloques fundamentales los cuales son secuenciales por lo cual no existe retroalimentación, es decir, el flujo de información se hace en un solo sentido tal cómo se observa en la Figura 2.9.

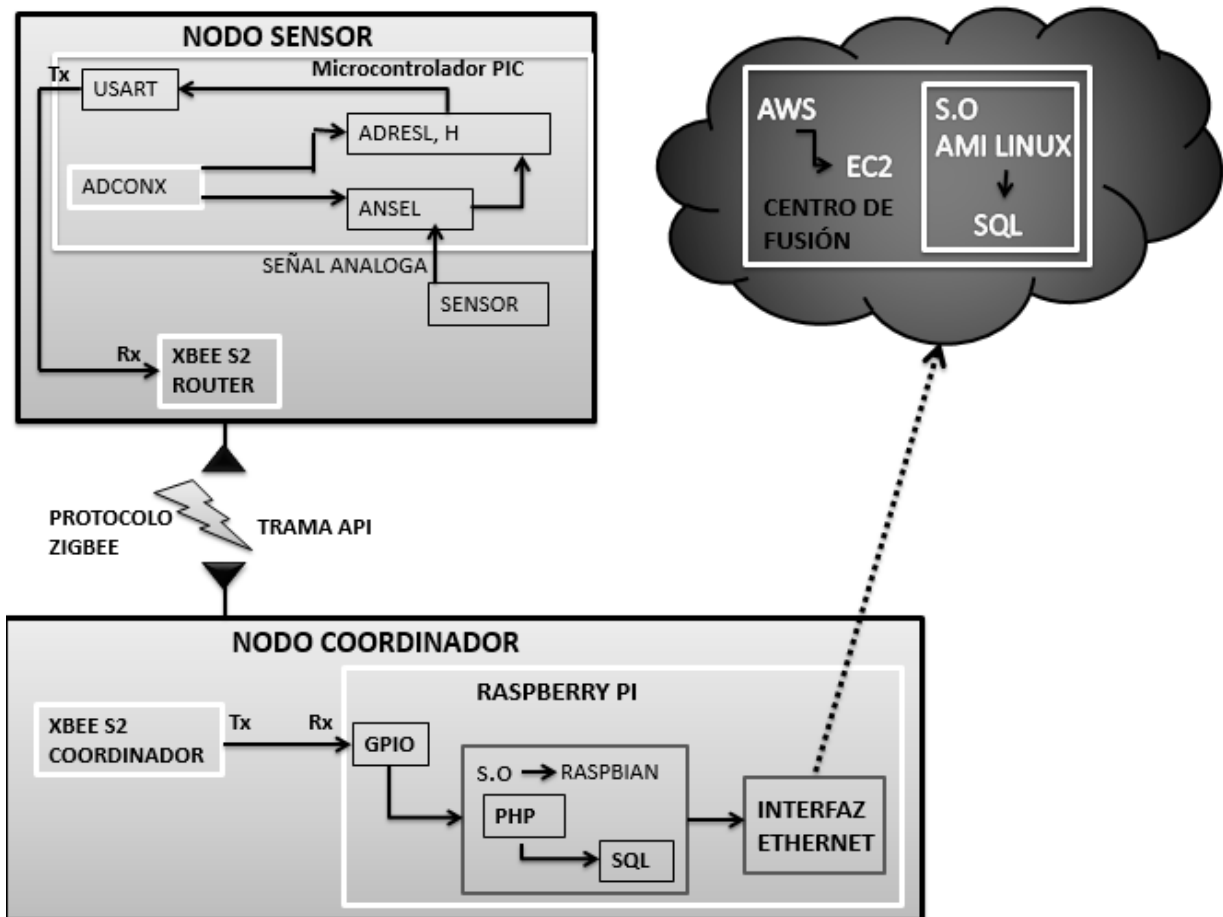


Figura 2.9: Diseño del prototipo.

El primer bloque está constituido por el nodo sensor, el cual se encargará de enviar diferentes datos de voltaje en forma inalámbrica hacia un nodo coordinador y este a su vez enviará dicha información a un servidor localizado en la nube. A continuación se describirán algunos elementos de estos bloques:

- **Nodo Sensor (Contiene tres elementos):** uno de los elementos de este bloque es el microcontrolador (elemento 1) el cual posee dos módulos utilizados para esta aplicación. El primero tiene que ver con la conversión análogo digital, el cual es configurable a través de los registros ADCON0 y ADCON1; la recepción de la señal análoga del sensor (elemento 2), se hace a través de los puertos controlados por el registro ANSEL y el resultado digital de la conversión es el registro ADRESL y ADRESH. El segundo módulo es el USART y tiene que ver con transmisión de los datos en forma serial al dispositivo transmisor XBEE serie2 (elemento 3).
- **Nodo coordinador (Contiene 2 elementos):** Uno de los elementos de este bloque es el Xbee serie2, el cual recibe la información por medio del protocolo Zigbee del nodo sensor. El segundo elemento es la “Raspberry Pi” el cual recibe la información del módulo Xbee serie2 a través de sus puertos GPIO, la cual será procesada por el sistema operativo (S.O) y posteriormente a través de un algoritmo en PHP se envía a una base de datos local SQL y a otra base de datos externa localizada en la nube por medio de una INTERFAZ ETHERNET con el protocolo Tcp/Ip.
- **Centro de Fusión:** El centro de fusión se encuentra compuesto por elementos de software un sistema operativo (S.O) y un servidor de base de datos (SQL).

El tipo de comunicación entre el nodo sensor y el nodo coordinador es inalámbrico a través del protocolo Zigbee, adicionalmente el nodo coordinador se comunica a través del puerto ethernet de la Raspberry Pi por medio del protocolo tcp/ip con el centro de fusión localizado en la nube. En este último aspecto, es posible reemplazar el puerto ethernet por una antena USB wifi (opcional), que se conectada en uno de los puertos USB2.0 de la raspberry pi y por medio de un router con acceso a internet asociado a esa antena se puede ingresar al centro de fusión.

2.5. Diseño del Transmisor

El diseño del transmisor consta de tres elementos: el primero, es una resistencia variable que toma la muestra de voltaje y la suministra al segundo elemento, que es el microcontrolador, el cual es el encargado de convertir esa señal análoga en una señal digital y al mismo tiempo transmitirla de forma serial a través de su pin TX. El último elemento es el módulo Xbee (configurado como router) el cual recibe la señal del micro en su pin RX para luego ser transmitida en forma inalámbrica al nodo coordinador. La Figura 2.10 representa las conexiones eléctricas.

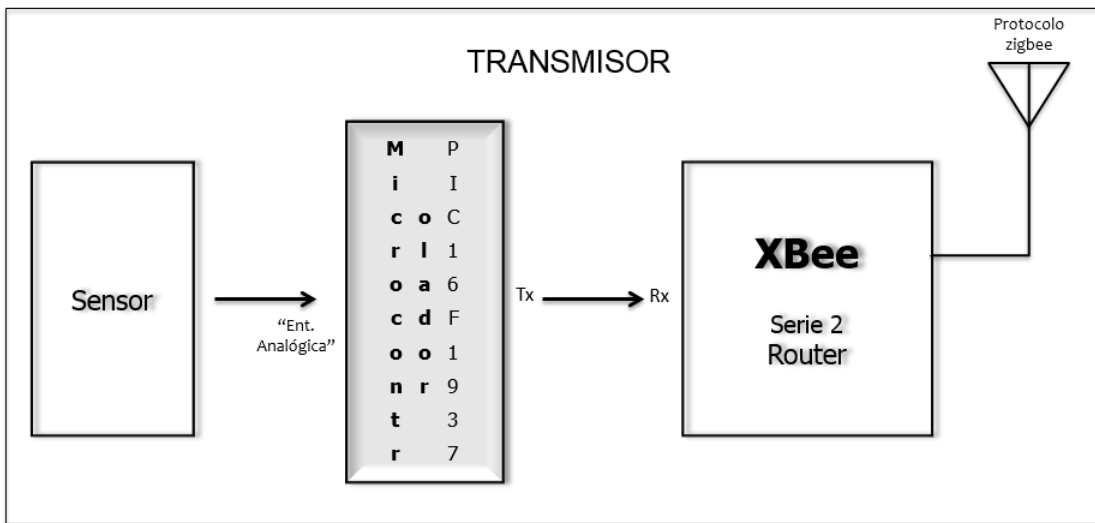


Figura 2.10: Diseño del Trasmisor.

2.6. Diseño del Receptor

El diseño del receptor [Figura 2.11] de datos resultaría de la siguiente forma: El módulo Xbee serie 2 conectado a la Raspberry Pi modelo B. Este dispositivo se encarga de la recepción de datos. Una vez que el módulo Xbee reciba la trama emitida por el nodo sensor, éste la transmite en su pin TX y es recibida por el pin RX de la Raspberry pi.

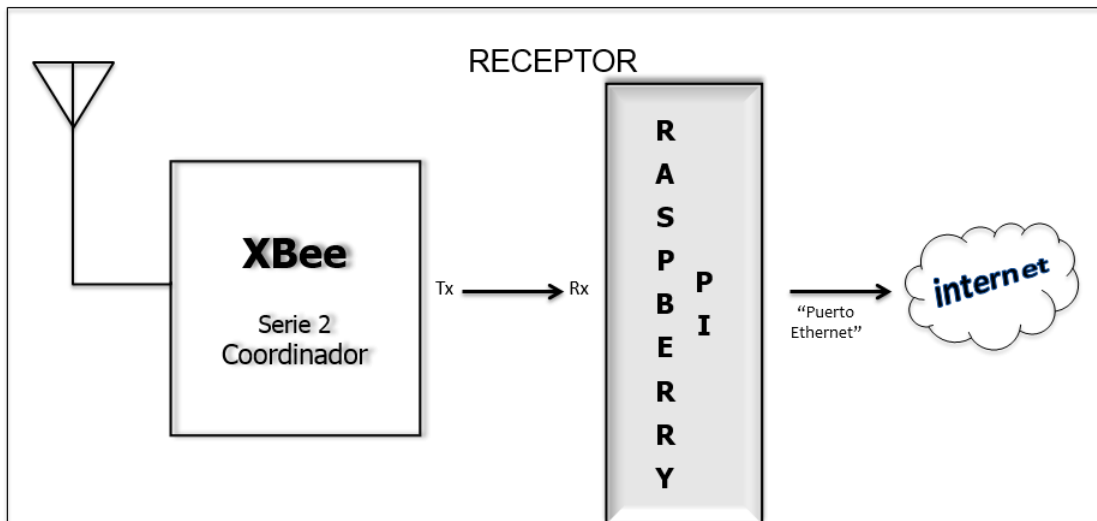


Figura 2.11: Diseño del Receptor.

Internamente la Raspberry procesa la información recibida a través de un registro del sistema llamado “ttyAMA0” el cual recibe los datos a una velocidad en Baudios (9600Bps) para luego almacenarla en la base de datos local. Dentro de este diseño se deja especificado que el nodo receptor o coordinador puede recibir las señales de varios nodos sensores debido a que el protocolo Zigbee permite el intercambio de información entre varios nodos o de varios nodos a uno por medio de las direcciones MAC.

2.7. Diagrama de Contexto y Casos de Uso.

El diseño de software se encuentra incorporado al manejo y tratamiento que se le está dando a la información cuando es recibida en el receptor por lo cual los casos de uso representan la descripción total del sistema, asociándolos a los actores que intervienen en él, para este diseño serán dos: el sistema y el usuario.

2.7.1. Casos de uso

En los casos de uso se establecerán las actividades que ejecutaran cada uno de los actores dentro del sistema, estos estarán acompañados de las tablas de especificación que permiten establecer una descripción clara de cada uno de los casos de uso.

a. Caso de uso 1. Ejecutar Script: El sistema de software en este caso el receptor conformado por el sistema operativo instalado en la Raspberry pi la Raspberry Pi recibe la información a través de un Script el cual posteriormente almacena la información en una base de datos local [Figura 2.12].

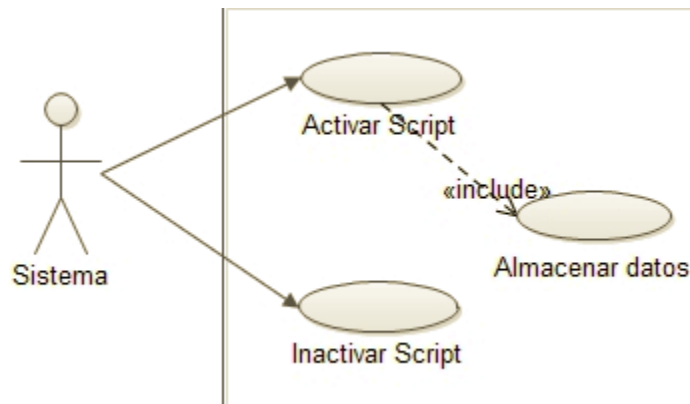


Figura 2.12: Diagrama Caso de Uso 1 “Ejecutar Script”.

La descripción del caso de uso se encuentra detallada en Tabla 2.8 en donde la ejecución del script debe ejecutarse para en forma autónoma sin que el usuario intervenga o en su defecto activarlo manualmente.

Tabla 2.8: Descripción del caso de uso número 1.

Especificación del caso de uso: Ejecutar script.	
Descripción:	Permite al Sistema activar e inactivar el script que permite la recepción y transmisión de los datos. Así mismo estos datos serán almacenados en una base de datos local.
Autor:	Adolfo Andrés Castro Sánchez
Actor1:	Sistema: es el software encargado de ejecutar el algoritmo que captura los datos recibidos del nodo sensor y los guarda en la BD, el cual opera desde el corazón del sistema operativo
Precondiciones:	El Sistema debe acceder a la consola de la Raspberry Pi.
Pos condiciones:	Activar Script.
Casos de uso que incluye:	Almacenar Datos

b. Caso de uso 2. Visualizar datos: Para el caso de analizar u observar los datos el usuario debe intervenir para realizar esta operación [Figura 2.13].

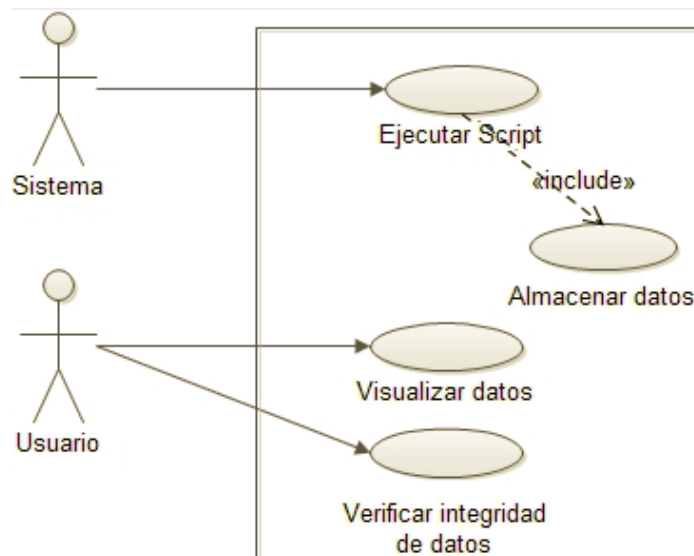


Figura 2.13: Diagrama Caso de Uso 2 “Visualizar los datos”.

Visualizar la información de los nodos sensores permite al usuario saber el estado de la información de las variables en cada uno de los nodos sensores [Tabla 2.9].

Tabla 2.9: Descripción del caso de uso número 2.	
Especificación del caso de uso: Visualizar Datos.	
Descripción:	Permite al usuario ver los datos de cada uno de los nodos y sus respectivos sensores los cuales se encuentran almacenados en la base de datos ubicada en el servidor de local o en el centro de fusión que se encuentra en la nube.
Autor:	Adolfo Andrés Castro Sánchez
Actor1:	Sistema: es el software encargado de ejecutar el algoritmo que captura los datos recibidos del nodo sensor y los guarda en la BD, el cual opera desde el corazón del sistema operativo
Actor2:	Usuario: Persona encargada de visualizar, imprimir y analizar los datos recogidos por el sistema, el cual opera desde la consola del sistema.
Precondiciones:	El Usuario deberá acceder al servidor local o el centro de fusión
Pos condiciones:	Observar los datos.
Casos de uso que incluye:	Seleccionar sensor.

2.8. Modelo Entidad Relación (MER).

La base de datos que se maneja dentro del sistema se representa mediante un modelo entidad relación (MER) la cual se está compuesta por tres tablas tal como se ilustra en la Figura 2.14.

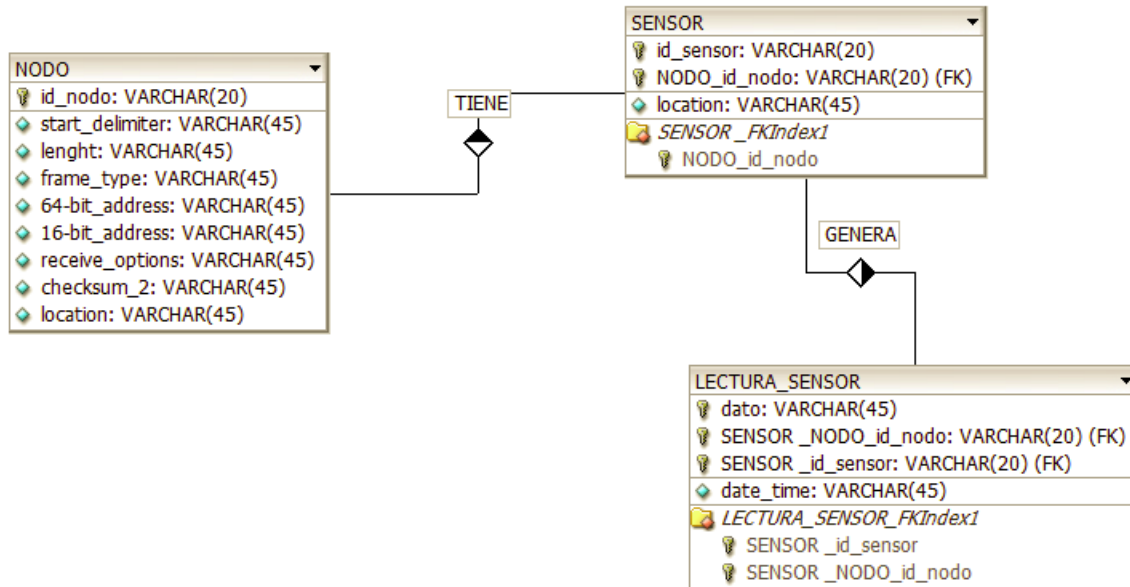


Figura 2.14: Modelo Entidad Relación de la base de datos.

La primera, es la tabla NODO que contiene 8 atributos y un identificador para poder saber qué nodo es el que se está visualizando y dicha tabla estará conectada a la segunda por medio de una relación de “uno a muchos”, lo que significa que la tabla nodo contiene muchos sensores. La segunda, es la tabla SENSOR que la cual tiene el atributo ubicación y un identificador para saber cuál de todos los sensores es. Esta tabla también ira conectada a una tercera tabla por medio de una relación “uno a muchos”, ósea que un sensor podrá tener varias lecturas. Por último la tercera tabla está compuesta por el atributo dato. Estos datos son facilitados la mayoría por la trama API generada transmitida desde el nodo sensor hasta el nodo coordinador sin embargo el geoposicionamiento es generado por un dispositivo GPS y la fecha es generada por el motor de la base de datos.

Capítulo 3

Dimensionamiento

El sistema diseñado anteriormente estará incrustado o embebido dentro de una red inalámbrica de sensores, para ello se debe estimar cuál es el alcance o dimensión de la red, sobre la cual su puesta en marcha no altera sus funciones originales es decir, disminuye la probabilidad bloqueo, son cortos los tiempos de latencia en la entrega de paquetes y tampoco pierde sus funciones naturales de implementación.

Para ello, se simulará la red a través de un software para estimar la cantidad de datos, tramas o símbolos que viajan a través del canal, antes de realizar la simulación se hará una breve descripción del estándar IEEE 802.15.4 sobre el cual se basa el protocolo Zigbee para conocer sus características de funcionamiento el cual se encuentra en el manual técnico de la tesis capítulo “*Estándar IEEE 802.15.4/Zigbee*”. Después, se hace el cálculo del “*link Budget*” para saber cuánto es el alcance y potencia de la antena para posteriormente describir el modelo del canal y su ecuación de pérdidas por trayectoria definiendo cuál es el radio de trabajo o área geográfica a cubrir. Luego, se exponen las ventajas y desventajas de trabajar con sincronización de nodos además se define la arquitectura de red. Para concluir el capítulo, se hará la simulación de la red para estimar el número de nodos que podrán ser instalados de acuerdo a la topología escogida.

3.1. Cálculo “Link Budget”.

Realizar y analizar el Link Budget¹ permite diseñar un enlace de comunicaciones para calcular un área de cobertura en red en condiciones de operación. Por lo tanto un sistema de comunicación por radio necesita diseños para proporcionen suficiente potencia recibida con el fin de asegurar que la relación mínima señal-ruido (SNR) sea superada todo el tiempo[27]. La Figura 3.1 hace la descripción de un esquema general para el *Link Budget*:

¹“Link Budget” es un cálculo que relaciona los factores de ganancia y de pérdida de asociado con las antenas, transmisores, líneas de transmisión y entorno de propagación utilizado para determinar la distancia máxima a la que un transmisor y un receptor puede con éxito operar.

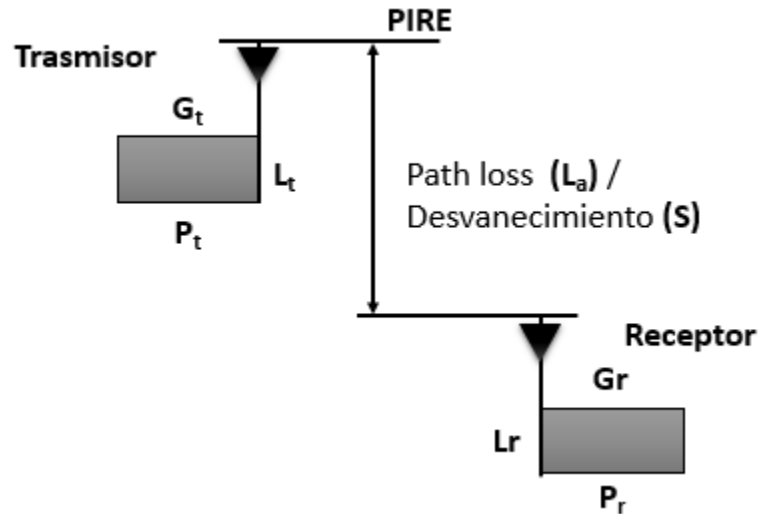


Figura 3.1: Descripción del Link Budget.

En donde:

P_t = Potencia de Salida del trasmisor (dBm o dBW)

P_r = Potencia de recepción (dBm o dBW)

L_t = Pérdidas del trasmisor por conectores (dB)

G_t = Ganancia de la antena de transmisión (dB)

L_a = Pérdida por trayectoria de espacio libre (dB)

S = Margen de pérdidas por desvanecimiento (dB)

Gr = Ganancia de la antena de recepción (dB)

L_r = Pérdidas del receptor por conectores (dB)

Con el fin de calcular el Link Budget para el enlace de radio Xbee, se arma la ecuación la Ecuación 3.1

$$P_r = (P_t - L_t + G_t) - (L_a + S) + (Gr - L_r)[dB] \quad (3.1)$$

La Potencia Isotrópica Radiada Equivalente (PIRE) puede ser expresada como:

$$PIRE = (P_t - L_t + G_t)[dB] \quad (3.2)$$

Ahora reemplazando en la ecuación 3.1 se obtiene lo siguiente:

$$P_r = PIRE - (L_a + S) + (Gr - L_r)[dB] \quad (3.3)$$

Dado que los módulos Xbee, que se seleccionaron para el enlace de radio consisten en chip integrado de antenas, no existen L_r y L_t . También se considera sin antenas de alta ga-

nancia por lo que no existe tampoco existen G_t y G_r . La Tabla 3.1² tiene consignados los parámetros a tener en cuenta para desarrollar la ecuación 3.1:

Tabla 3.1: Parámetros de las antenas Xbee S2 Digi .

Parámetros	Recepción	Transmisión
Potencia de salida del trasmisor (P_t)	0 dBm / -30dBW	
Ganancia de la antena (G_t, G_r)	0 dBi	0 dBi
Pérdidas por desvanecimiento (S)	10 dB	
Potencia mínima de recepción (P_r)	-92dBm / -122dBW	
Frecuencia de los módulos Xbee	2.4 GHz	

Como:

$$PIRE = (P_t - L_t + G_t)[dBW]$$

$$PIRE = (-30 - 0 + 0)[dBW]$$

$$PIRE = -30dBW$$

Para calcular las Pérdidas por trayectoria en espacio libre (L_a) en dB.

$$P_r = PIRE - (L_a + S) + (G_r - L_r)$$

$$L_a = PIRE - S + G_r - L_r - P_r$$

$$L_a = -30 - 10 + 0 - 0 - (-122)$$

$$L_a = 82dB$$

Pérdidas por trayectoria en espacio libre (L_a) también puede ser calculado usando la siguiente Ecuación 3.4³:

$$L_a = 10 \log_{10} \left(\frac{4\pi f d}{c} \right)^2 [dB] \quad (3.4)$$

Donde:

L_a = Pérdida por trayectoria de espacio libre (dB)

d = Distancia entre el trasmisor y receptor (m)

f = Frecuencia de la señal (Hz)

c = Velocidad de la luz (m/s)

Reemplazando en la Ecuación 3.4, (L_a) puede expresarse como:

²Extraídos de la ficha técnica de fabricación de los módulos Xbee (Digi International).

³Fórmula extraída de la ecuación de pérdidas por trayectoria de espacio libre de Harald T. Friis

$$La = 10\text{Log}_{10} \left(\frac{4\pi fd}{c} \right)^2$$

$$82 = 10\text{Log}_{10} \left(\frac{4(3,14)(2,4e^9)d}{3e^8} \right)^2$$

$$10^{82/10} = \left(\frac{4(3,14)(2,4e^9)d}{3e^8} \right)^2$$

$$158,48e^6 = 10106,48d^2$$

$$d = \sqrt{15696,95}$$

$$d = 125,28m$$

De acuerdo al análisis del Link Budget se puede decir que: asociando a la topología en estrella del diseño propuesto para este trabajo, se puede tener un radio de cobertura de hasta 125 metros sin embargo para dejar márgenes escalables se trabajará con un radio de 100 metros sin obstáculos y con línea de vista.

3.2. Modelo del Canal

El modelo de canal es una referencia utilizada para calcular las pérdidas por trayectoria de un enlace inalámbrico, mientras más estén alejados el transmisor del receptor mayor serán las pérdidas. El presente trabajo toma como referencia la utilización del dispositivo que captura los datos de nodos sensores y los carga en forma directa a un centro fusión en un ambiente urbano, para ello se hablara un poco de trabajos similares realizados con enlaces inalámbricos con dispositivos Xbee serie 2 estudiados en el capítulo 2. En Tailandia las empresas dedicadas a servicios de venta de energía eléctrica tratan mucho el tema de “*Smart Grid*”⁴ para aprovechar todas las facilidades que puedan ofrecer las redes eléctricas no obstante la

⁴se puede definir como la integración dinámica de los desarrollos en ingeniería eléctrica y los avances de las tecnologías de la información y comunicación (o TIC), dentro del negocio de la energía eléctrica; permitiendo que las áreas de coordinación de protecciones, control, instrumentación, medida, calidad y administración de energía, etc., sean concatenadas en un solo sistema de gestión con el objetivo primordial de realizar un uso eficiente y racional de la energía.

infraestructura ofrece muchos servicios dentro de estas redes pero hace falta mucho todavía sobre todo en el tramo final o “*Last mile*” en donde se interactúa con el consumidor [28]. Una de las cosas que más se requiere es la parte de Telemetría en donde la plataforma de Zigbee juega un papel muy importante sin embargo para este trabajo solo se centrará en la parte del modelo de canal en donde del trabajo anteriormente mencionado se hizo un experimento en donde se utilizaron dispositivos Zigbee con las siguientes características referidas en la Tabla 3.2 las cuales son muy parecidas a las utilizadas en este trabajo.

Tabla 3.2: Características de los dispositivos Zigbee utilizados.

Parámetro	Símbolo	Valor
Banda de Operación	BW ISM	2.4 GHz
Velocidad	Rb	250 kbps
Potencia de transmisión	PTx	-25 dBm to 0.6 dBm ajustable
Potencia de Recepción	RSSI	-90 dBm

El experimento arrojó los resultados de la gráfica Figura 3.2 Por lo cual entre más alejado este el transmisor del receptor las pérdidas de la señal son mucho mayores.

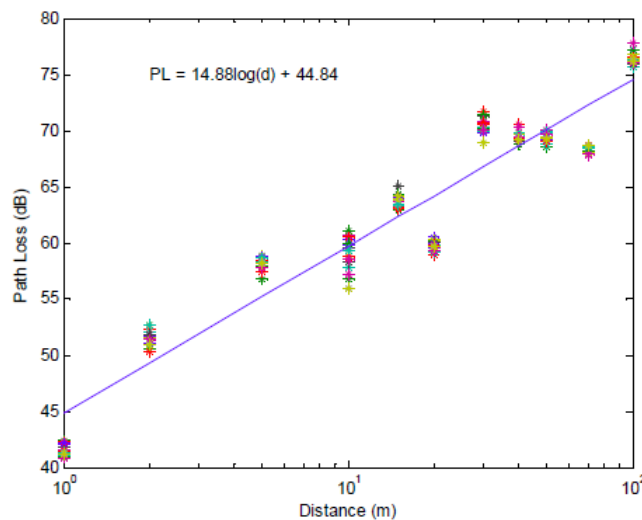


Figura 3.2: Grafica perdidas de trayectoria en un escenario abierto. [28]

La ecuación de pérdidas en la trayectoria en este caso es:

$$Pl = 14,88 \log(d) + 44,84 \tag{3.5}$$

En donde:

Pl: pérdida por trayectoria

d: Distancia entre transmisor y receptor dada en metros

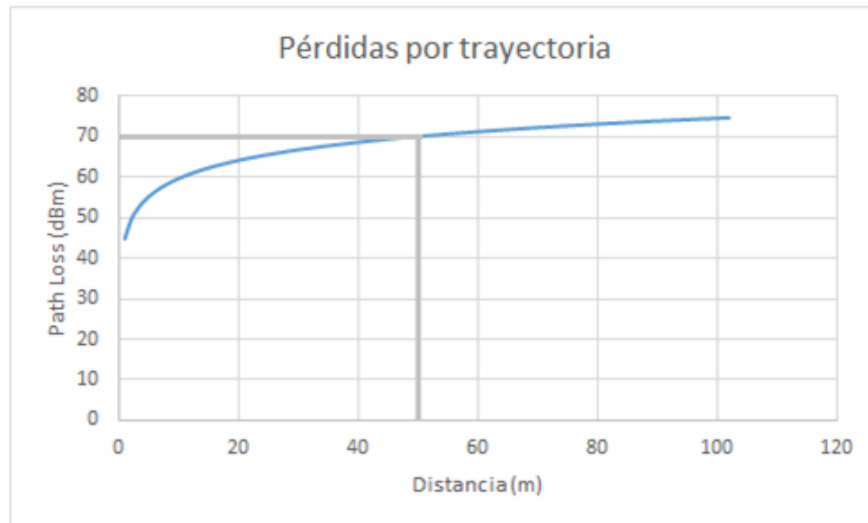


Figura 3.3: Pérdidas por trayectoria.

De acuerdo con el modelo de pérdidas por trayectoria y la ecuación 3.4 se puede inferir que el criterio utilizado para realizar este enlace es un radio de 50 metros [Figura 3.3] para ambientes urbanos en donde las pérdidas son aproximadamente de 70dB con la condición de que la potencia de radio del transmisor sea de mínimo de -20dBm, garantizando así un enlace confiable. La topología, por ahora, sería en estrella quedando tentativamente con 8 nodos router tal como lo ilustra la Figura 3.4.

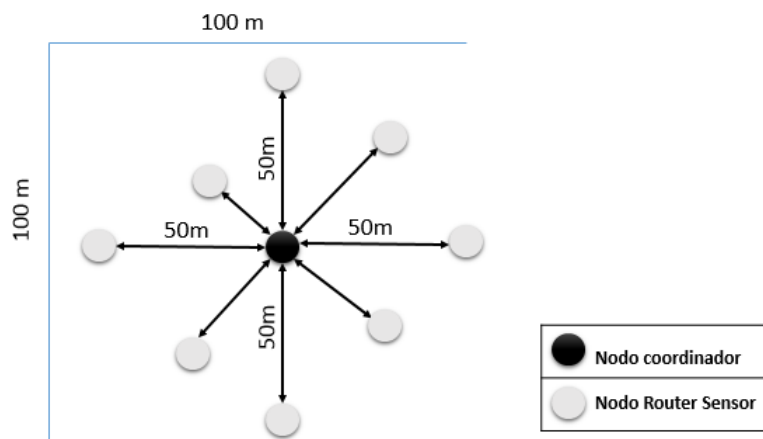


Figura 3.4: Radio de cobertura para una topología en estrella.

3.3. Sincronización de los nodos

Los siguientes son algunos de los factores que afectan la sincronización de nodos de redes Zigbee que se deben tener en cuenta al momento de implementar la red [29]:

- **Fallas de reloj:** Saltos repentinos en el conteo del tiempo provocando variaciones en frecuencia y saltos de tiempo.
- **Retardo asimétrico:** Debido a que en la comunicación inalámbrica de los nodos sensores se pueden tener diferentes retardos en el trayecto de ida y el de regreso de la información.
- **Temperatura:** Debido al despliegue aleatorio de los nodos sensores en varios lugares del terreno, la variación de temperatura en diferentes instantes pueden causar aceleración o retardo del reloj interno de los nodos sensores.
- **Ruido en frecuencia:** Por lo general este tipo de ruido se debe a la inestabilidad de la señal de reloj proporcionada por un cristal.
- **Ruido de fase:** Algunos de los causantes del ruido de fase pueden ser la fluctuación de acceso en una interfaz a nivel de hardware, la variación de respuestas del sistema operante ante interrupciones, el jitter en el retardo de la red, además se puede deber a la técnica de acceso al medio y retardos por encolado de los datos.

Las técnicas de sincronización deben superar estos obstáculos, y también ser conscientes de que las baterías de los nodos sensores son limitadas.

Entre estas técnicas se tiene:

- **NTP (*Network Time Protocol*):** Se basa en servidores temporales que deben ser a la vez legibles desde la red, robustos ante fallas y altamente precisos. La desventaja del uso de este protocolo es que algunos nodos para llegar a los servidores temporales necesitan pasar a través de otros nodos, mismos que pueden fallar.
- **EBS (*Refered Broadcast Synchronization*):** Básicamente el tiempo se traslada salto a salto desde el inicio o fuente de la difusión hasta el fin de la red. Dado que tras múltiples saltos se puede tener variaciones diversas, luego de la difusión se realiza comunicación entre nodos para transmitir offsets estimados.
- **TDP (*Time Diffusion Protocol*):** Permite comunicar el tiempo en la red con cierta tolerancia, misma que se puede ajustar dependiendo del propósito de la red de sensores inalámbricos. El protocolo de difusión de tiempo se autoconfigura eligiendo nodos maestros para sincronizar la red de sensores. Los nodos maestros se eligen tomando en cuenta los requerimientos energéticos y la precisión de los relojes.

La sincronización de nodos es una opción que posee ventajas y desventajas, para ello el mismo protocolo Zigbee tiene posibilidad habilitarla o no. “Un coordinador tiene la opción de transmitir señales “*beacon*” para sincronizar los dispositivos conectados a él. Esto es llamado una PAN con beacon habilitado (*beacon enabled PAN*). La desventaja de usar beacon es que todos los dispositivos en la red deben despertar de forma regular, para escuchar el beacon, sincronizar sus relojes y volver a dormir. Esto significa que muchos de los dispositivos en la red pueden despertar solamente para la sincronización y no desempeñar otra tarea mientras está activos. Por lo tanto, la duración de la batería de un dispositivo en una red con beacon habilitado es normalmente menor que una red sin beacon” [30].

3.4. Arquitectura de red para el dimensionamiento

En redes en estrella el coordinador de red, llamado coordinador PAN, se sitúa en el centro, y toda conexión que se quiera realizar entre los distintos nodos de la red debe pasar por éste. Éste nodo central activo dispone de los medios necesarios para prevenir problemas, ya que por él circulan todos los paquetes de las distintas conexiones entre módulos. En la Figura 3.5 vemos ilustrada la típica estructura de una red en estrella.

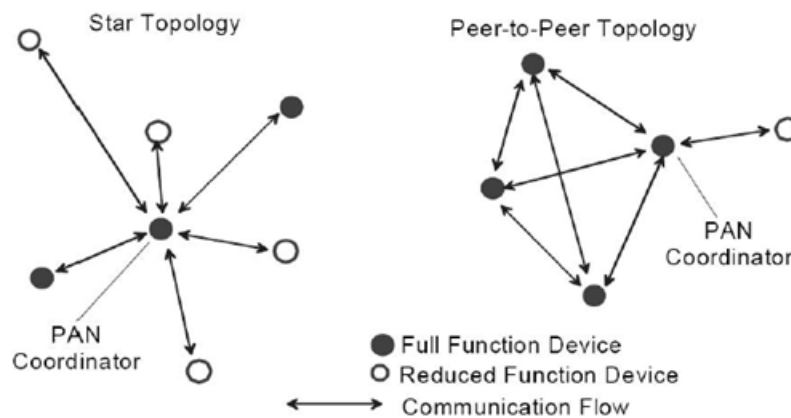


Figura 3.5: Topologías de red contempladas en la norma IEEE 802.15.4.

La alimentación de los dispositivos finales puede ser por baterías mientras que el coordinador debe estar directamente conectado a través de la red eléctrica ya que su consumo es mucho mayor.

Las ventajas que nos puede aportar una red en estrella es la facilidad a la hora de implementarla, adecuada para redes temporales, el fallo de un nodo periférico no influiría en el comportamiento del resto de la red y no hay problemas con colisiones de datos ya que cada estación tiene su propia conexión al coordinador central. En contrapartida, la utilización de una red en estrella nos limita el número de nodos que pueden estar conectados a la red.

También se debe tener muy en cuenta que los costes de mantenimientos pueden aumentar a largo plazo, y que el fallo del nodo central puede echar abajo la red entera [31].

Otro problema de esta configuración es que la expansión de la red está muy limitada puesto que el rango de alcance del coordinador es el que define el tamaño de la red. A causa de todo ello, podemos confirmar que una red en estrella puede ser poco fiable en el momento de realizar transferencias de información.

3.5. Dimensión general de una red Zigbee

Una red Zigbee puede tener hasta 65535 dispositivos interconectados entre si y divididos en subredes de hasta de 255 nodos en teoría se podrían tener en una topología en estrella hasta 255 nodos sensores y un coordinador. Sin embargo, es un modelo ideal puesto que en la práctica esta implementación tendría muchos problemas en todas las capas y el tráfico de red podría ser incontrolable aumentando los problemas que se estudiaron en el capítulo uno.

3.6. Enlace inalámbrico para el dimensionamiento

De acuerdo a lo estudiado en la descripción del estándar 802.15.4 para la banda 2.4Ghz utilizando una técnica de modulación DSSS con codificación O-QPSK con acceso a cualquiera de los 16 canales a través de CSMA-CA se tiene una velocidad de transmisión de los datos de 250Kbps. La Figura 3.8 analiza la composición de una trama a nivel de capa física.

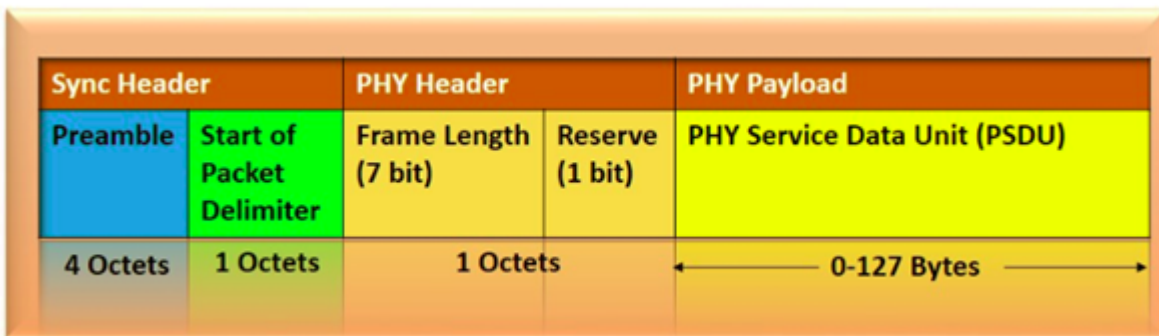


Figura 3.6: Composición de una trama física en el estándar 802.15.4

Como se puede observar la trama más larga en este protocolo es de aproximadamente 133 bytes divididos en Preámbulo (4), delimitador (1), longitud (1) y carga útil (0-127).

Si 133 bytes son 1064 bits, por lo tanto si:



Se puede decir que una trama de 133 bytes en el canal se podría transmitir en 4.256ms lo cual es relativamente rápido si la aplicación no exige velocidad en la transmisión de datos. Por lo tanto el canal de trasmisión se ocuparía solo en un:



El canal se estaría ocupando en un 0.4 % del total de su capacidad. Si tenemos en cuenta un criterio de bloqueo del canal está entre el 70 % y 100 % la probabilidad de bloque es muy baja. por otro lado la teoría habla de un mínimo en la tasa de transmisión de 20Kpbs, realizando el mismo ejercicio anterior se tiene:

Si 133 bytes son 1064 bits Por lo tanto si:



Se puede decir que una trama de 133 bytes en el canal se podría transmitir en 53.2ms lo cual se convierte en una velocidad muy lenta, si la aplicación exige velocidad en la transmisión de datos esta referencia generaría muchos problemas de latencia. Por lo tanto el canal de transmisión se ocuparía solo en un:



El canal se estaría ocupando en 5 % del total de su capacidad. Si tenemos en cuenta un criterio de bloqueo del canal está entre el 70 % y 100 % la probabilidad de bloqueo sigue siendo muy baja pero de igual forma el retardo es muy alto.

Los ejercicios anteriores no tienen en cuenta fenómenos físicos o efectos multitrayectoria que hacen que el canal tenga desvanecimiento.

3.7. Simulación y análisis de una red Zigbee

Las condiciones de simulación de la red están especificadas en el reporte técnico de la tesis en el capítulo “Estándar IEEE 802.15.4/Zigbee” sección “condiciones de simulación” Tabla .

Tabla 3.3: Parámetros de simulación Topología en estrella.

Estándar o Norma	IEEE 802.15.4
Técnica de Modulación:	DSSS (direct sequence spread spectrum)
Banda:	2.4GHz
Capas a Analizar:	PHY y MAC
# Nodos coordinadores	1
# Nodos Router	20 hasta 2
Tráfico de red	Solo al Coordinador
Tamaño de la tramas	50 bytes
Tiempo de Simulación:	10 minutos

3.7.1. Resultados de la simulación

Para analizar los resultados se deben recordar cuales fueron los parámetros de simulación: 14 paquetes por minuto con tamaño de 50 bytes (400 bits) (criterio que conlleva a la instalación de un número mayor de sensores en cada nodo sensor cubriendo grandes áreas), de 20 a 2 nodos con topología en estrella enviando esta información en forma simultánea al nodo coordinador. El número de nodos escogidos tiene que ver con el parámetro de una subred Zigbee que acepta un número máximo de 255 nodos por lo cual se escogió un criterio del 10%; de esos nodos se restaron 5 como margen de corrección (criterio) por ello el número de nodos escogidos fue de 20. De acuerdo La Tabla 3.4 contiene los resultados obtenidos en la simulación de acuerdo a 7 escenarios (para: 20, 17, 14, 11, 8, 5 y 2 nodos).

Tabla 3.4: Resultado de la simulación en diferentes escenarios.

Parámetros	Número de Nodos							Unidades
	20	17	14	11	8	5	2	
Throughput	82000	81000	79000	77000	57000	36000	14300	Bps
Trafico Recibido Coordinador	192000	177000	160000	130000	84000	55000	18200	Bps
Retardo	12,5	11,8	9,5	8,2	6,1	4,5	3,5	ms
Carga de la Red	205000	173000	140000	105000	59000	44000	14100	Bps

De acuerdo con esta tabla se puede decir que el tráfico generado por el aumento en el

número de nodo es bastante significativo y afecta las condiciones de operación de la red. La gráfica de la Figura 3.7 corresponde a la carga de datos de toda la red, teniendo en cuenta que la carga la establecen los nodos router enviando paquetes al coordinador.

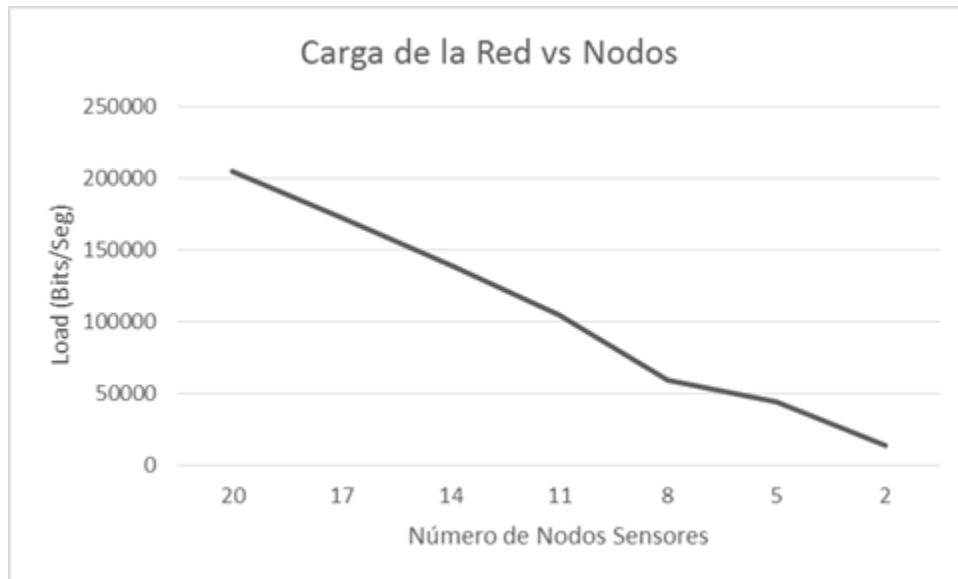


Figura 3.7: Grafica carga de la Red Vs # Nodos

La gráfica de la Figura 3.8 corresponde al retardo generado por los paquetes que fueron entregados desde su origen hasta el coordinador.

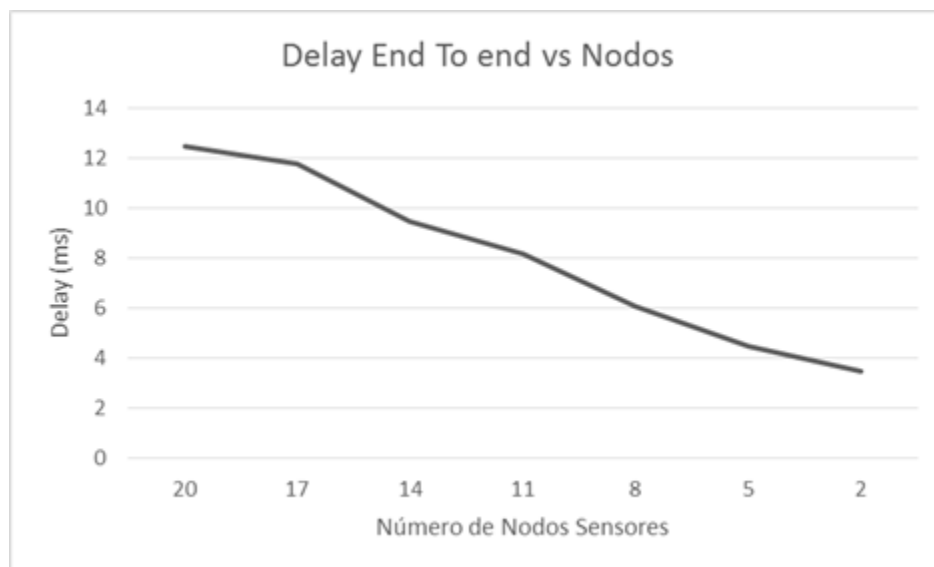


Figura 3.8: Grafica Delay end to end Vs # Nodos.

La gráfica de la Figura 3.9 corresponde al tráfico recibido por el coordinador, es muy importante esta gráfica porque es la permite estimar cuanto sería el tráfico de red en cada escenario en donde varía únicamente el número de nodos.

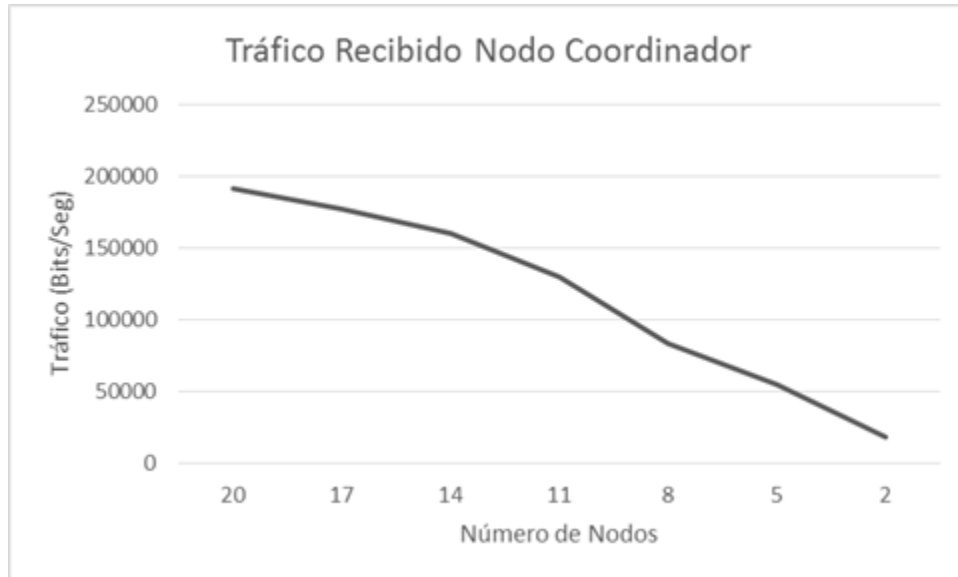


Figura 3.9: Grafica tráfico recibido Nodo Coordinador.

La gráfica de la Figura 3.10 corresponde al Throughput vs el número de nodos.

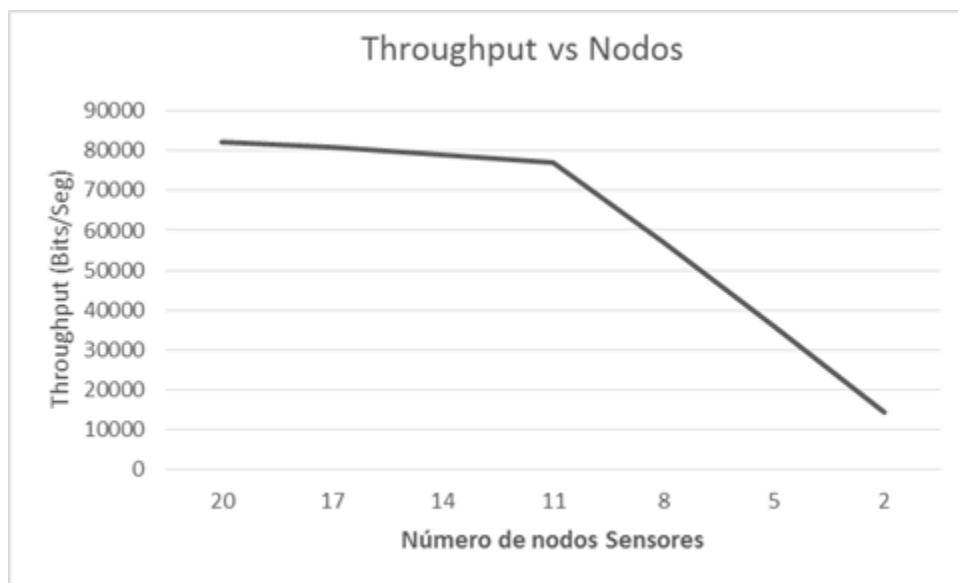


Figura 3.10: Grafica Throughput Vs Nodos.

De acuerdo al análisis de estas gráficas y los resultados generados por el simulador, se puede determinar tiempo de latencia de los paquetes para una red estrella conociendo el tráfico de cada nodo con la siguiente ecuación:

$$Vl = \frac{\sum_{n=1}^{\infty} Vn}{K} \tag{3.6}$$

En donde:

Tl: Tiempo de latencia de todos los paquetes para un número N de nodos (s)

Vn: Tráfico de todos los paquetes para el mismo número de N nodos (Bps)

K: capacidad del canal (Bits)

n: Número de nodos

Sin embargo es posible omitir la sumatoria, si solo si, el tráfico de cada uno de los nodos del escenario correspondiente es constante, por lo cual quedaría de la siguiente forma:

$$Tl = \frac{Vn(total)}{K} \tag{3.7}$$

La Tabla 3.7 ilustra los tiempos en que tardan en transmitir todas las tramas de todos los nodos en cada uno de los escenarios (se asume que el tráfico en cada nodo es el mismo en todo el escenario correspondiente).

Tabla 3.5: Tiempo de transmisión de los paquetes.

Nodos (N)	Vn (total) (Bps)	Capacidad del Canal (K)					
		250Kb	200Kbps	150Kbps	100Kbps	50Kbps	20Kbps
		Tl	Tl	Tl	Tl	Tl	Tl
Para 20 =	192000	0,77s	0,96s	1,28s	1,92s	3,84s	9,60s
Para 17 =	177000	0,71s	0,89s	1,18s	1,77s	3,54s	8,85s
Para 14 =	160000	0,64s	0,80s	1,07s	1,60s	3,20s	8,00s
Para 11 =	130000	0,52s	0,65s	0,87s	1,30s	2,60s	6,50s
Para 8 =	84000	0,34s	0,42s	0,56s	0,84s	1,68s	4,20s
Para 5 =	55000	0,22s	0,28s	0,37s	0,55s	1,10s	2,75s
Para 2 =	18200	0,07s	0,09s	0,12s	0,18s	0,36s	0,91s

La Tabla 3.5 puede explicarse a través de la gráfica de la Figura 3.11.

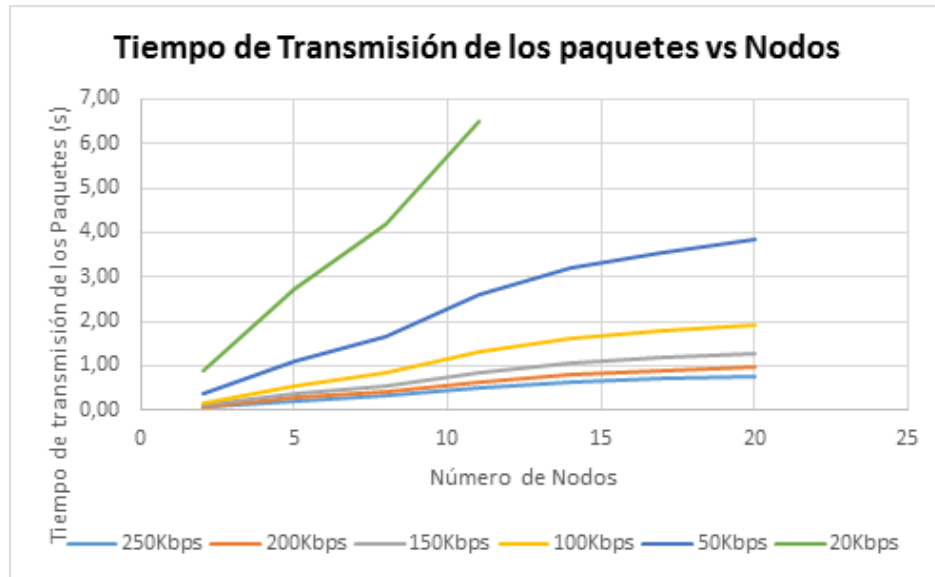


Figura 3.11: Gráfica Tiempo de transmisión de los paquetes Vs # Nodos.

En donde los colores representan la velocidad (Kbps), el eje “x” representa los escenarios de acuerdo al número de nodos y el eje “y” representa el tiempo en se demoran en transmitir todos los paquetes de acuerdo a la velocidad y al número de nodos, de la anterior gráfica se toma como criterio escoger un número de 8 nodos de tal manera que con una velocidad de 150 Kbps el tiempo de entrega de los paquetes es de 0,56s.

3.8. Propuesta para el dimensionamiento de la red.

Tabla 3.6: Criterio para el dimensionamiento de una red Zigbee (Estrella).

Criterios de Dimensionamiento	
Nodos	Coordinador (1) y Router (8)
Tamaño del paquete	50 Bytes
Paquetes por segundo por nodo	14
Trafico en bits según de la red	84Kbps
Capacidad del canal	170Kbps (Criterio)
Ocupación del canal	49%
Radio de cobertura	50 metros (Criterio)

Teniendo en cuenta los criterios de la Tabla 3.6, se propone que el dimensionamiento de la red Zigbee para este trabajo sea de una red distribuida con topología en estrella conformada por el nodo central (Coordinador) y 8 nodos sensores (Router). Se puede decir que el dimensionamiento de la red es de 8 nodos sensores por cada nodo coordinador, enviando 14 paquetes por segundo cada uno del tamaño de 50 bytes si la velocidad de transmisión es de 170Kbps [Figura 3.12].

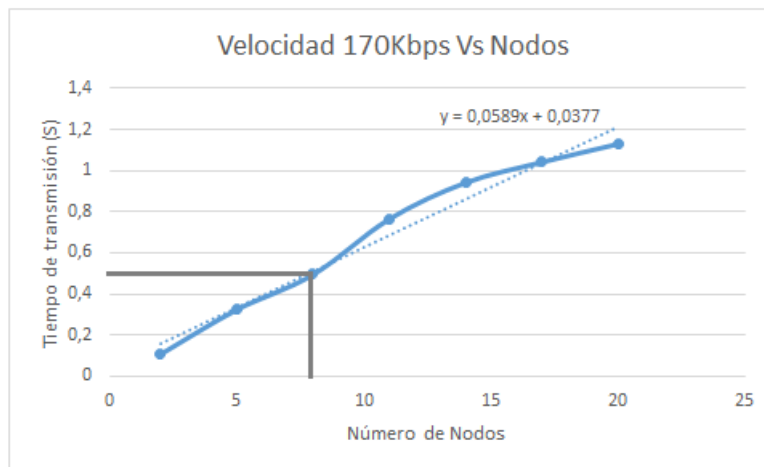


Figura 3.12: Grafica de velocidad 170Kbps Vs # Nodos.

Los nodos coordinadores estarán conectados a la nube de internet, por lo cual la concentración de datos que normalmente hay en una topología “*mesh*” o “*tree*” desaparecen, disminuyendo la “Latencia”, la red queda abierta para sea para que sea más escalable y trabajaría muy eficazmente si el tráfico es menor al propuesto, además no siempre se está utilizando toda la capacidad del canal.

Capítulo 4

Implementación

Implementar un dispositivo que capture los datos de una red de sensores y los cargue en forma directa a la nube, involucra recursos tecnológicos de hardware, software, diseño, consulta con expertos, recomendaciones de los fabricantes, entre otros. La construcción del dispositivo se estudiará en este capítulo involucrando los elementos de hardware, software, programación de los dispositivos físicos y ejecución de los algoritmos de funcionamiento. Tanto la implementación de software como la de hardware se explicarán a detalle analizando todos y cada uno de los pasos utilizados en el desarrollo, recordando que los recursos tecnológicos utilizados fueron estudiados en el capítulo uno y luego en el capítulo dos se escogieron los dispositivo que finalmente se utilizaron para la implementación, no obstante elementos escogidos no representan la única forma de diseño posible por lo tanto su elección tienen que ver por el tema de los costos, documentación, manejo en la industria, versatilidad, disponibilidad, etc. Para ello los elementos más representativos son:

- Módulos de comunicación Inalámbricos Xbee S2
- Ordenador de placa única «*Raspberry Pi modelo B*»
- Microcontrolador PIC16F1937
- Sistema Operativo Debian
- Servidor Web Lightpd
- Base de datos MySQL

Construir este mismo dispositivo con otros recursos tecnológicos podría incrementar el presupuesto tres veces más de lo que costó este, puesto que de acuerdo a lo averiguado a nivel comercial de los elementos expuestos en el capítulo uno y dos, éstos son los más favorables del mercado. Los elementos y algoritmos utilizados son de fácil abstracción de tal manera que se pueden utilizar para trabajos futuros u otros proyectos.

4.1. Implementación del diseño de hardware

4.1.1. Esquema de Implementación

La Figura 4.1 describe en forma general el diseño estudiado en capítulo 2 pero en fase de implementación.

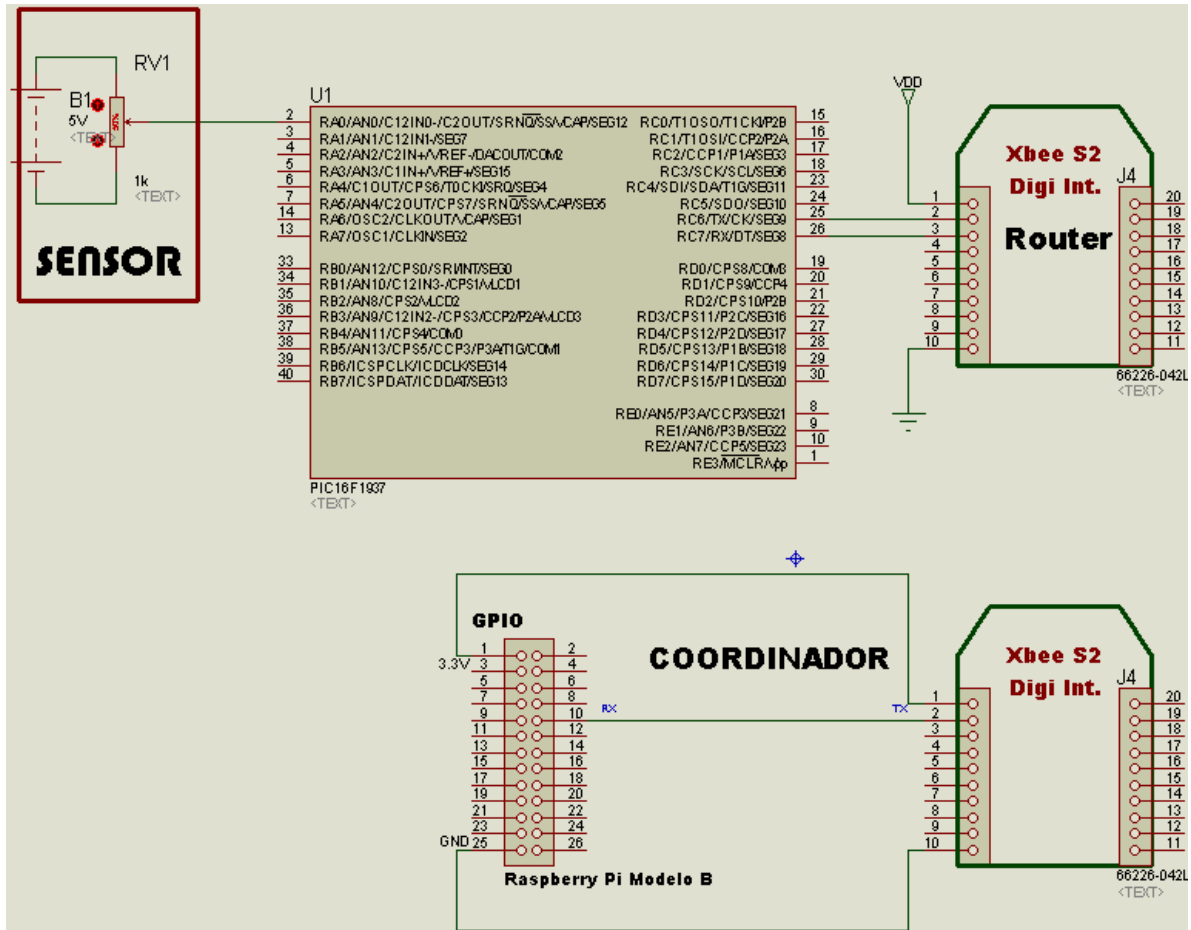


Figura 4.1: Diseño general normalizado

4.1.2. Implementación del Diseño.

La tarjeta que contiene el microcontrolador es un prototipo de prueba [Figura 4.2] ya que no corresponde al diseño de implementación final pero sirve como referencia para la pruebas hechas antes de la prueba final.

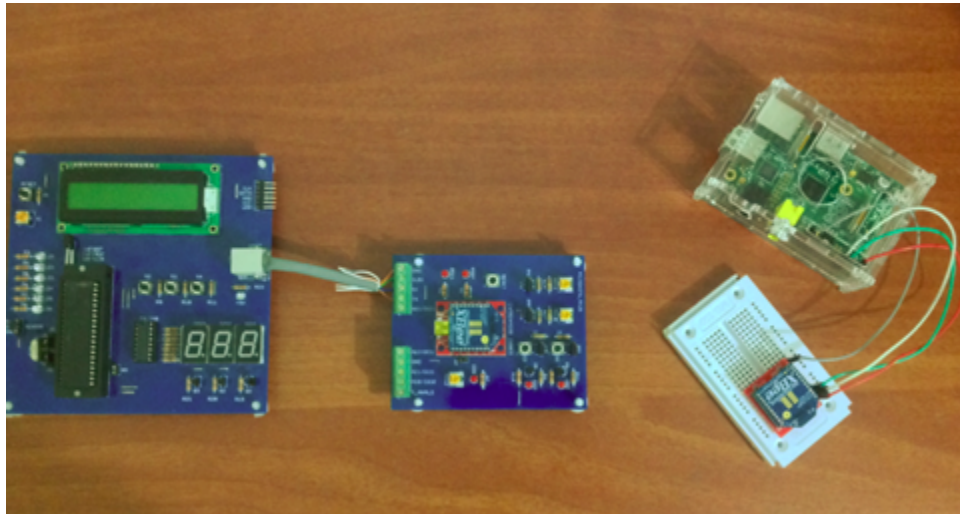


Figura 4.2: Diseño general implementado en pruebas.

4.1.3. Implementación por bloques

a. El primer bloque [Figura 4.3] es el del nodo sensor:

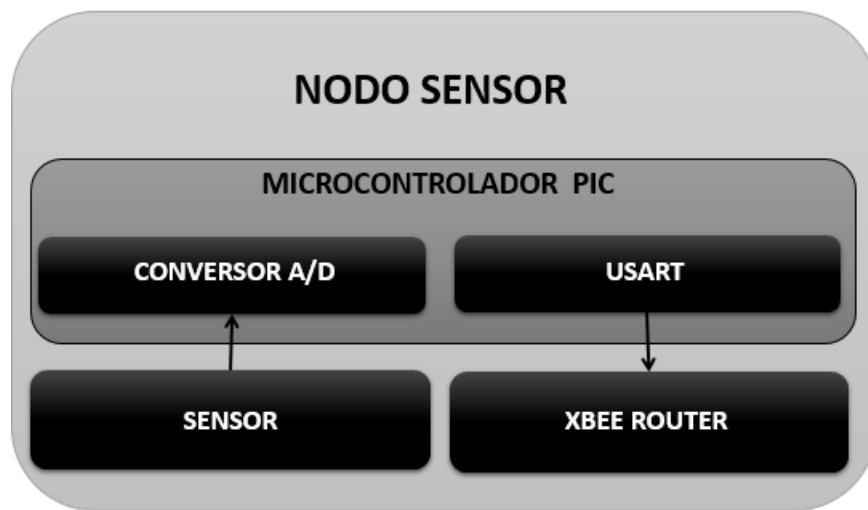


Figura 4.3: Estructura por bloques del nodo sensor.

En este nodo es posible conectar hasta 14 sensores pero la implementación de este dispositivo sugiere conectar una resistencia variable la cual enviará datos de voltaje al nodo coordinador.

b. El segundo bloque [Figura 4.4] corresponde al nodo coordinador

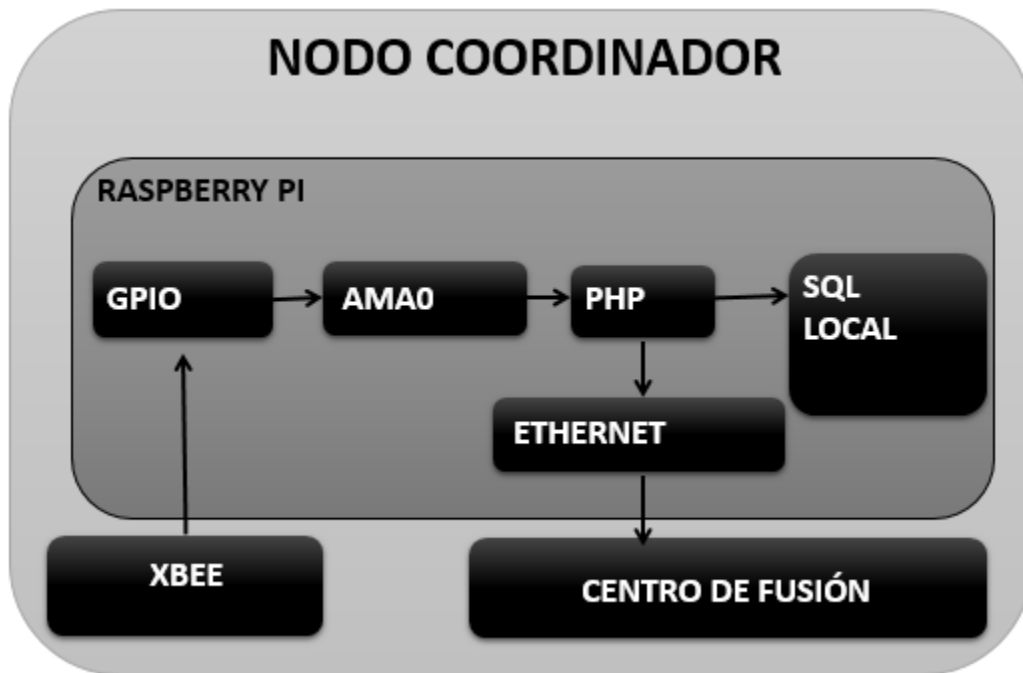


Figura 4.4: Estructura por bloques del nodo coordinador.

En el nodo coordinador se destaca como elemento central el ordenador de placa reducida el cual es el encargado de procesar la información recibida en forma inalámbrica.

c. El tercer bloque [Figura 4.5] es llamado centro de Fusión



Figura 4.5: Estructura por bloques del centro de fusión.

Este bloque contiene únicamente elementos de software, los cuales se tratarán en el capítulo 5.

4.1.4. Implementación del Nodo Coordinador

a. En esquemático [Figura 4.6] o diseño normalizado

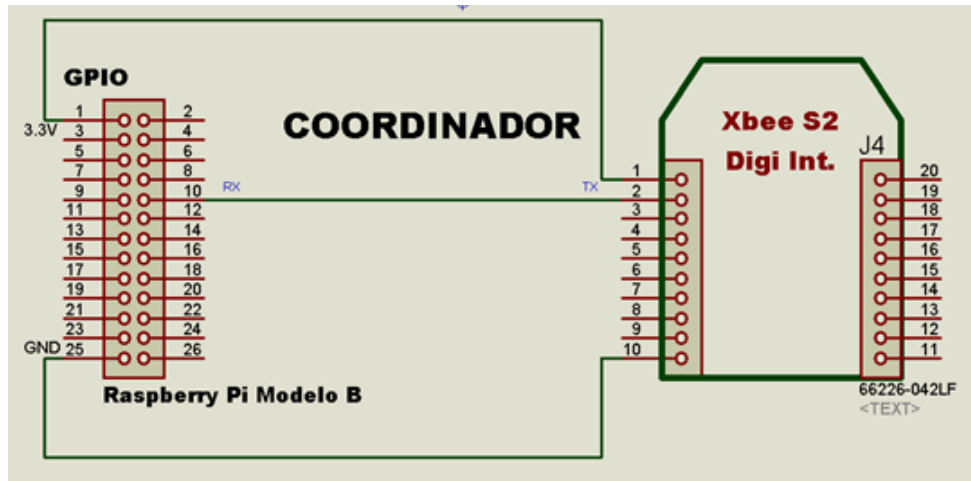


Figura 4.6: Diseño normalizado del nodo coordinador.

b. Diseño de prueba [Figura 4.7] implementado.

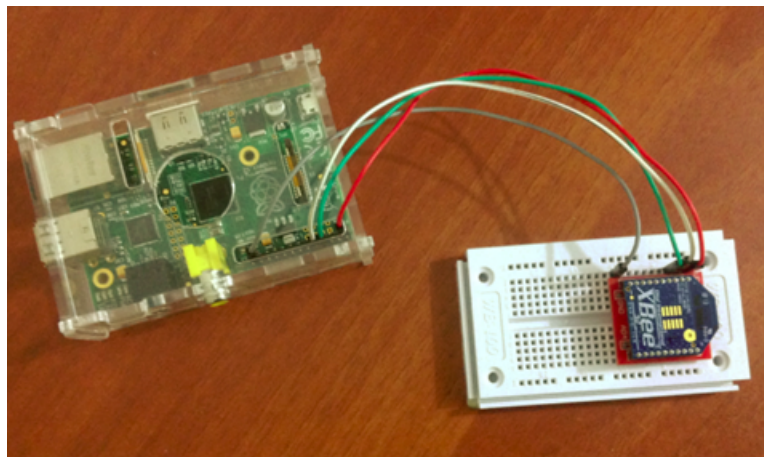


Figura 4.7: Diseño implementado del nodo coordinador.

Hay que anotar que el módulo Xbee trabaja a 3,3V sin embargo como la Raspberry pi en su pi GPIO #1 tiene la posibilidad de alimentar dispositivos externos a 3,3V por lo tanto en este caso no existe ningún problema de acople, es decir se puede hacer en forma directa.

4.1.5. Implementación del Nodo Sensor

a. En esquemático [Figura 4.8] o diseño normalizado.

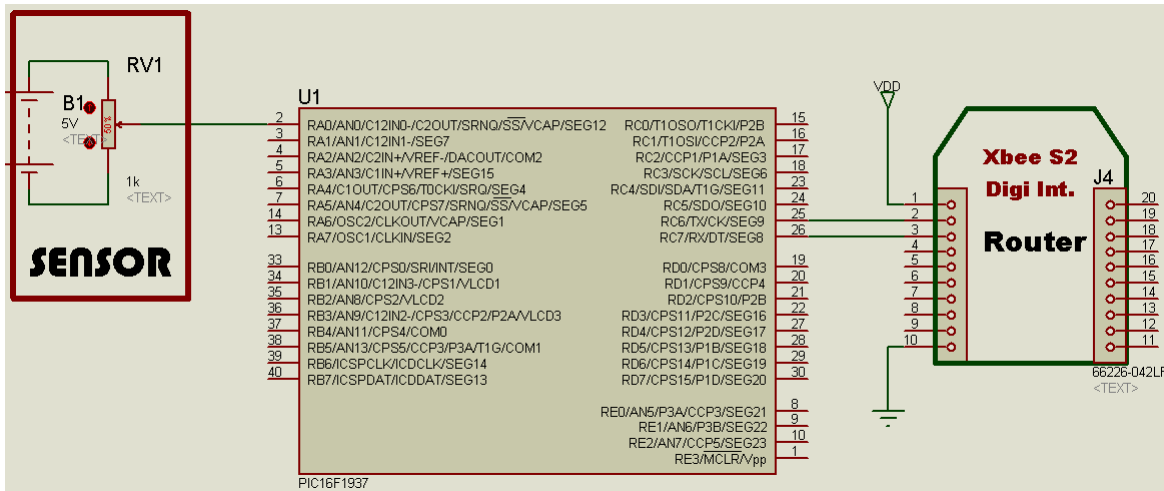


Figura 4.8: Diseño normalizado nodo sensor.

b. Diseño de prueba [Figura 4.9] implementado

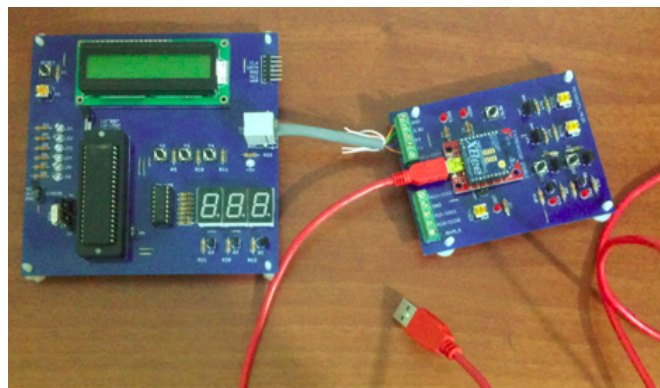


Figura 4.9: Diseño implementado en pruebas del nodo sensor.

Al igual que en el paso anterior, en este se debe tener en cuenta el tema de los voltajes de alimentación de los dispositivos. Para el caso del microcontrolador PIC debe ser previamente configurado en sus “bits configuration” para que no se reinicie cuando este alimentado por debajo de 4,0V. Los detalles técnicos de implementación se encuentran consignados en el reporte técnico de la tesis el se encuentra como anexo de la misma así como también recomendaciones y sugerencias al momento de implementar (capítulos: “Configuración de la Raspberry Pi”, “Configuración de los módulos Xbee”, “Programación del Microcontrolador

PIC 16F1937”, “Análisis de la trama API emitida y recibida”, “Configuración del módulo UART de la Raspberry” y “Crear la base de datos, la tabla y los campos”). Sin embargo a continuación se presenta de manera resumida algunos detalles de la implementación.

4.2. Implementación del dispositivo

Los aspectos a tener en cuenta son a la hora de implementar el dispositivo previamente diseñado son:

1. Algoritmo de para la recepción de los datos en el nodo sensor (o Router), el siguiente pseudocódigo es el que se ejecuta al interior de microcontrolador para tomar la muestra del sensor y posteriormenete enviársela al Xbee router:

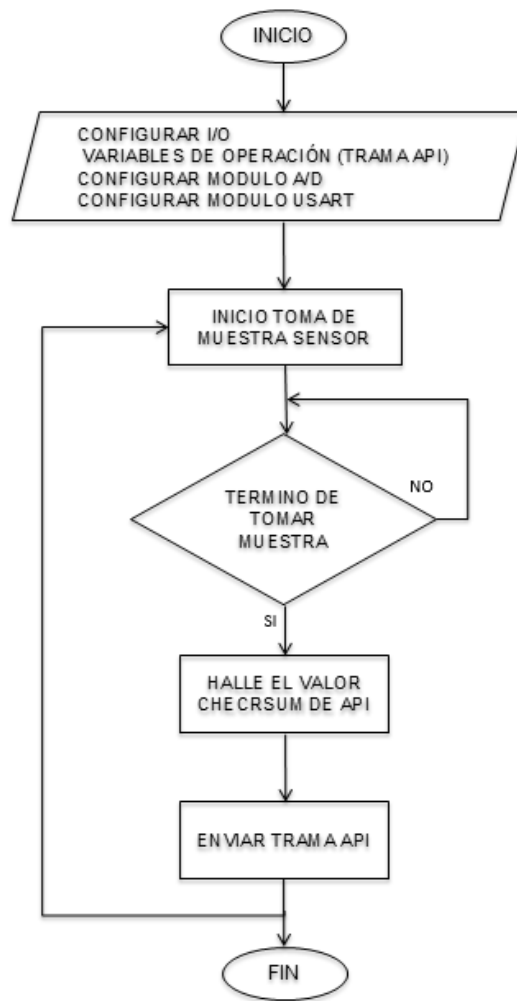
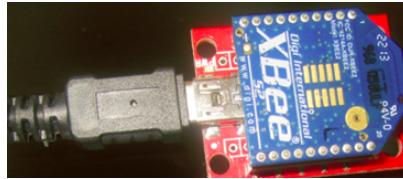


Figura 4.10: Diagrama de Flujo del algoritmo en el Micro

2. Programar los Módulos Xbee para ser configurados como Router y Coordinador Respectivamente



Módulo y programador Xbee.

3. Configurar La Raspberry Pi



Accediendo a la Raspberry mediante cliente ssh.

4. Crear la base de datos:

```
pi@raspberrypi ~ $ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 53
Server version: 5.5.38-0+wheezy1 (Debian)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database BD_sensores1;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| BD_sensores1 |
| base_prueba |
| mysql |
| performance_schema |
| phpmysqladmin |
| talbasedatos |
+-----+
8 rows in set (0.00 sec)

mysql> use BD_sensores1;
Database changed
mysql>
```

5. Crear el algoritmo de ejecución en PHP para la recepción de los datos.

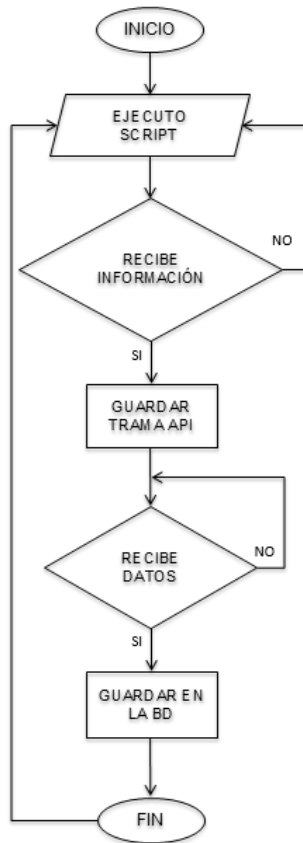


Figura 4.11: Algoritmo en PHP para la recepción de los datos

Los anteriores aspectos están mucho más detallado en el reporte de técnico de tesis adjunto a este documento.

Transmisor:



Receptor:



Una vez encendidos los dispositivos se ejecuta el script php en la consola de la Raspberry. A continuación se ofrecen las imágenes de los dispositivos trabajando en forma simultánea y con conexión a internet:



Los problemas de integridad de los datos van asociados al tema de la seguridad, por esta razón, se recomienda utilizar un algoritmo que verifique o traduzca la trama recibida en el servidor central antes de ingresarla a la base de datos, de tal manera que no se ingresen datos desordenados o información errada. Para terminar con este numeral, se procede a verificar que la información haya sido guardada en la base de datos en AWS:

4.3. Niveles o capas del sistema implementado.

La propuesta de implementación se podría resumir en niveles o capas, para este trabajo la propuesta de niveles es la siguiente:

Tabla 4.1: Niveles o capas de la aplicación realizada.

MySQL (Local)	
PHP – Cgi	
“ttyAMA0”(Kernel)	CAPA MAC (Pi)
GPIO 15 (Pi)	Ethernet
802.15.4/Zigbee	TCP/IP
Microcontrolador	Servidor AWS
Sensor Análogo	MySql(EC2-AWS)

Los niveles físicos están determinados por conexiones eléctricas y protocolos de comunicación, el resto de niveles están asociados al software y algoritmos de recolección de datos. Estos niveles son la propuesta materializada de los objetivos planteados en este trabajo.

Capítulo 5

Centro de Fusión

Las ciudades inteligentes se caracterizan por poseer las plataformas de hardware y software necesarias para brindar el soporte a los usuarios y organismos de inteligencia para la atención y prevención de desastres. De las plataformas de hardware que se encuentran enviando información constantemente son las redes inalámbricas de sensores no obstante como se ha estudiado en capítulos anteriores éstas poseen problemas de implementación los cuales pueden ser mitigados de muchas formas de tal manera que se reduzcan los inconvenientes asociados a la latencia y escalabilidad sin embargo en este capítulo se abarcará todo lo relacionado con el tema del centro de fusión.

El centro de fusión es un lugar en donde la información estará almacenada y dispuesta a ser analizada, en el capítulo 4 se pudo implementar un centro de fusión local, es decir, los datos están almacenados en el ordenador que se encuentra recibiendo la información en primera instancia no obstante esta información debe cumplir con ciertas características sobre todo a nivel de seguridad. De acuerdo con esto en este capítulo se analizarán plataformas especializadas en guardar datos de redes de sensores en la nube, así como sus ventajas y desventajas. Una vez analizadas estas plataformas se elegirá de ellas, en la cual se implementarán los elementos de software necesarios para guardar las muestras recogidas por el dispositivo ya implementado. Cuando se tengan estos elementos se procede a analizar el tema de la conexión a internet para acceder al centro de fusión localizado en la nube, para ello se describirá la trama de la variable a sensar, si se requiere sensado rápido o sensado lento y el tipo de tecnología de banda ancha.

Teniendo los principios de seguridad de la información en cuanto a integridad, disponibilidad y confidencialidad de los datos [32], se propone almacenar los datos recogidos a través del dispositivo en un servidor web. Para ello existen varias opciones dedicadas solo a guardar datos de sensores en la nube. A continuación se estudiarán varias de esas plataformas.

5.1. Plataformas que guardan datos para WSN

5.1.1. “Cloud Device”

Es una plataforma de almacenamiento en la nube en la cual se podrá conectar cualquier dispositivo y ensamblar en ambas direcciones. Estas son algunas de las ventajas:

- a. La capacidad de hablar con un módulo Xbee específico de una red situada en el otro lado del mundo para realizar operaciones personalizadas.
- b. Almacenar datos enviados desde diferentes módulos Xbee y recuperar más tarde.
- c. La configuración del dispositivo remoto.



Figura 5.1: Plataforma device cloud

[33]

El primer paso para la conexión de los módulos Xbee a dispositivos de la nube es la creación de una cuenta en “Cloud Device”. “Los pasos a seguir son: configurar el hardware y el software que conectará su red Xbee con internet y los dispositivos de la nube. Existen varias combinaciones de hardware + software disponible, pero todas las combinaciones necesitan tener una interfaz Xbee con acceso a la nube Figura 5.2 El más común es el uso de una pieza

de “Digi ConnectPort X Gateway” con un programa de Python personalizado o un PC con el software Xbee Internet Gateway” [34]:



Figura 5.2: Elementos para subir los datos de un Xbee a la nube a través de Cloud.
[33]

5.1.2. Plataforma “Xively”

La compañía *LogMeIn*, más conocida por sus aplicaciones de acceso remoto, tiene un papel relevante en la Internet de las Cosas. Tiene el servicio llamado “Xively” como una evolución de COSM (y antes Pachube).



Figura 5.3: Esquema general de conexión a Xively.
[35]

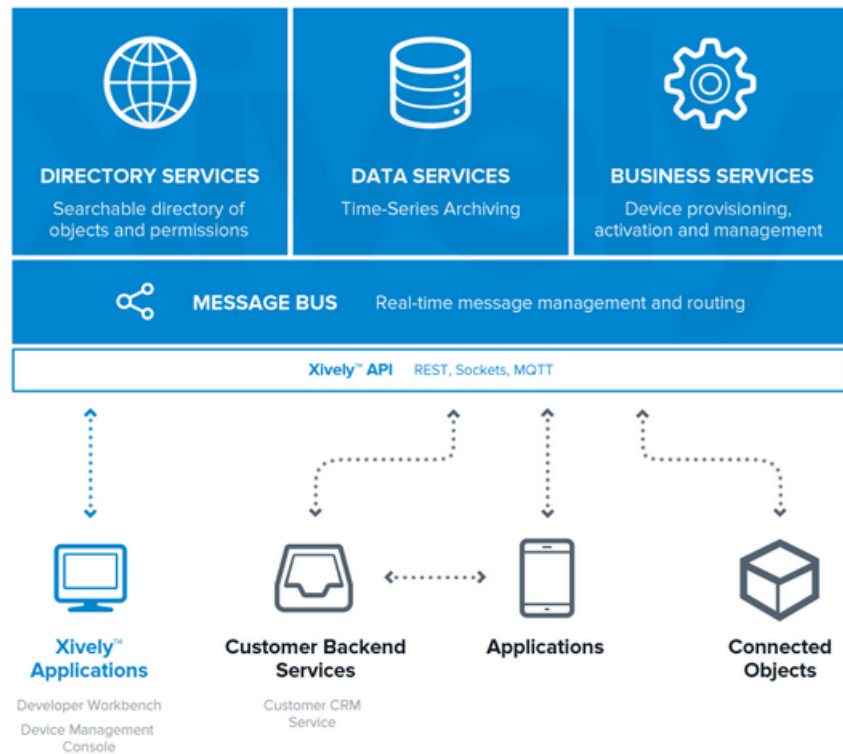


Figura 5.4: Flujo de los datos hacia Xively.
[36]

Xively se define como una “Plataforma como un Servicio” (PaaS) para la Internet de las Cosas. Esencialmente *Xively* es una nube especializada en recibir y desplegar información de los distintos sensores que se necesiten conectar [Figura 5.4]. Las principales características [37] de este servicio son:

- a. Herramienta para desarrolladores
- b. Conexión con servidores.
- c. Centro de aprovisionamiento.
- d. Servicio comercial orientado a Internet de las Cosas.

5.1.3. ODK Sensor

Otra alternativa de servidor en la nube es el framework de la plataforma *Open Data Kit* llamada “ODK Sensor” [Figura 5.5], la cual es una herramienta de código abierto con un

conjunto de elementos que ayuda a la solución y administración de recopilación de datos móviles con sistema operativo Android.



Figura 5.5: Aplicación ODK sensor para Android.
[38]

“ODK Sensores simplifica el desarrollo de aplicaciones móviles basadas en sensores mediante la creación de un controlador común que permita a todos los sensores acceder a través de una interfaz de detección unificado. Esta sola interfaz trae consigo sensores externos (USB, Bluetooth), así como sensores incorporados de Android a través de una interfaz común. La conexión de un sensor externo requiere un controlador especial de sensor que transforma los datos detectados en un formato que es fácilmente utilizable por las aplicaciones porque maneja threading y amortiguación, lo cual permite a los desarrolladores de *drivers* para sensor, centrarse en escribir piezas mínimas del código específico del sensor. Mediante la creación de una aplicación de sensor con componentes del controlador de sensor separados esperamos permitir un ecosistema de controladores de sensores compatibles y reutilizables” [39].

5.2. Centro de Fusión

Las plataformas estudiadas anteriormente son especializadas en servicios orientados a guardar datos de sensores sin embargo es necesarios disponer de un conjunto de servicios web cada vez más amplios y con un nivel de seguridad muy alto. Se debe contar con:

- Un servidor web
- Disponer de un servidor de base de datos
- Tener copias de seguridad - Servidores de inteligencia
- Potencia y dimensionamiento en un servidor de producción

Hoy en día, disponer de ese tipo de servicios en los proveedores de Hosting es una tarea complicada, puesto que se debe pagar cantidades iniciales de puesta en marcha y papeleos

interminables. Todo esto con el “*Amazon Web Service*” es más sencillo [40].

a. Disponer de un servidor Web: Con el servicio Amazon EC2 se puede levantar una instancia de servidor Linux o Windows en unos 15 minutos.

b. Disponer de un servidor de base de datos: Tener todo tipo de bases de datos y que se tengan disponibles en cuestión de minutos.

c. Envío de miles de correos con seguimiento de envío: Enviar miles de correos, sin las limitaciones de los servidores SMTP de otros proveedores.

d. Copias de seguridad, disponibles de forma inmediata: Hacer un backup desde un servidor de Windows es tan sencillo como crear un contenedor y utilizar para hacer el enlace con el contenedor de S3 de Amazon.

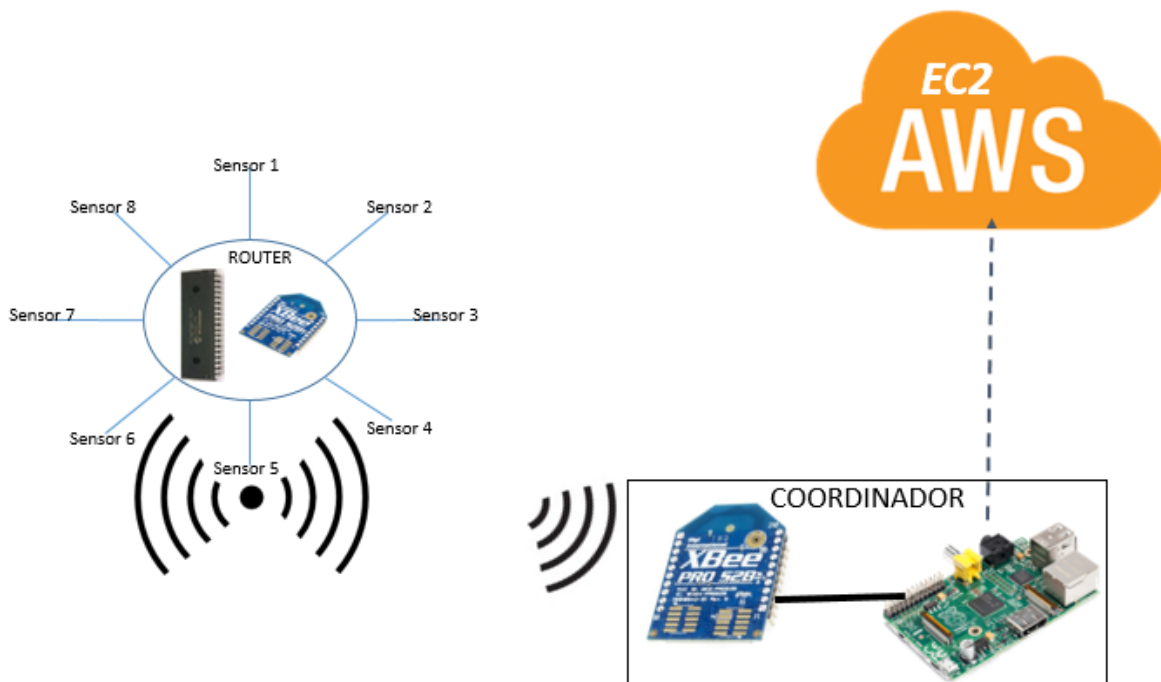


Figura 5.6: Centro de Fusión.

Por estas razones los servicios de Amazon prestan flexibilidad, seguridad y escalabilidad requisitos necesarios para futuros proyectos. Por lo tanto se escoge “*Amazon Web Service*” como el centro de fusión para almacenar los datos en la nube que serán enviados desde el nodo coordinador [Figura 5.6]. Los detalles técnicos y de implementación del centro de fusión y del dispositivo en funcionamiento se encuentran en el reporte técnico de la tesis en el capítulo “*Gestión en AWS*”.

5.3. Flujo de información

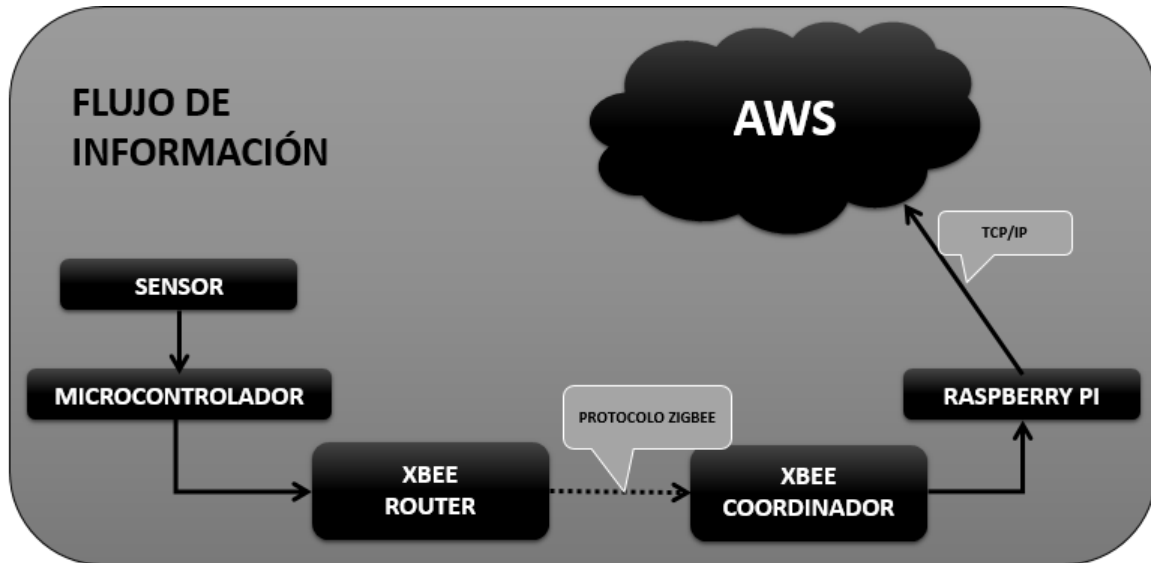


Figura 5.7: Flujo de Información.

Lo planteado en este capítulo recoge los elementos necesarios para implementar un centro de fusión localizado en la nube con los datos que se adquieren a través de los dispositivos, lo cual tiene muchas ventajas: disponibilidad de los datos en cualquier momento, integridad de la información, seguridad de la información, entre otros. Como se mencionó anteriormente, la plataforma de servicios en la nube de “amazon web services” fue seleccionada porque posee los elementos y herramientas necesarias para desarrollar proyectos que tengan que ver con sistemas autónomos o inteligentes, por ejemplo si se desea que el sistema detecte automáticamente un evento de extrema urgencia, esta pueda tomar automáticamente una decisión que atienda transitoriamente dicha emergencia.

5.4. Velocidad del enlace al centro de fusión

El centro de fusión se encuentra localizado en un servidor en *Amazon web services* por lo tanto para acceder a él es necesario tener conexión a internet, tal como lo sugiere la Figura 5.8 la cual ilustra 8 nodos sensores conectados a un nodo coordinador el cual posee una conexión a la nube, dicha velocidad aún es desconocida .

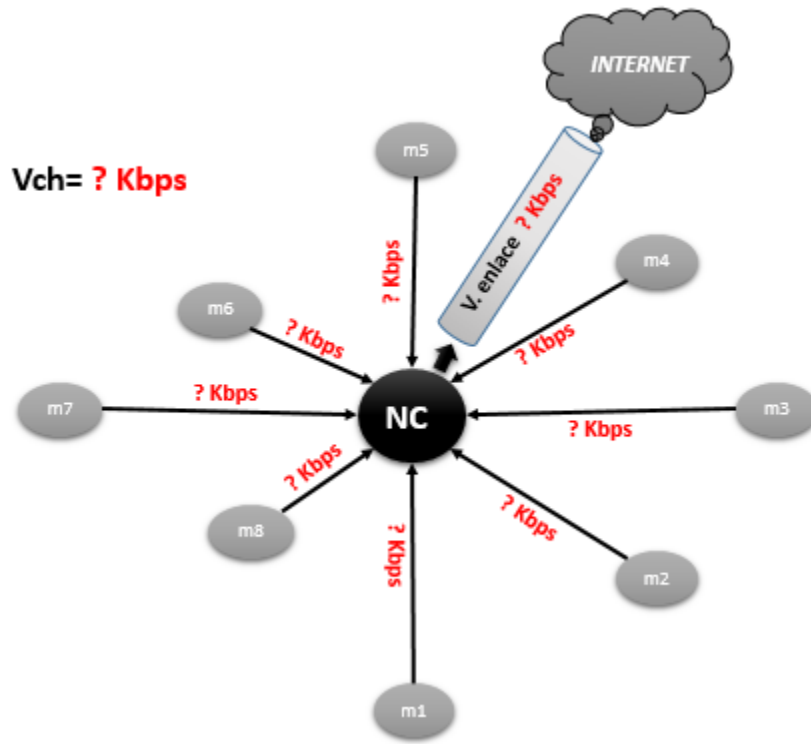


Figura 5.8: Tráfico de datos y conexión a la nube.

Los nodos *router* o *end device* (m_i) se encuentran asociados al nodo coordinador el cual tendrá acceso a internet a través de la *Raspberry pi*, por lo cual, los parámetros que se tienen en cuenta a la hora de establecer el tráfico de datos y la conexión a la nube son los siguientes:

$$V_{ch} = \sum_{i=1}^{N_{\text{nodos}}} [m_i] * \text{overhead} \quad (5.1)$$

En donde se tiene:

V_{ch} es la velocidad del canal colector

N_{nodos} = número de nodos asociados al nodo central.

NC = nodo central o nodo coordinador.

m_i = tráfico en bits de un nodo.

$Overhead^1 = 20\%$.

Ejemplo: Si se desea realizar el cálculo exacto para sensar la variable temperatura, se desarrolla de la siguiente forma:

Si $m_i =$ para variables [Temperatura]

Para armar la trama que contiene la variable temperatura se asignará una variable X_i para cada campo los cuales salen de la trama Zigbee y el campo generado por el motor de la base de datos:

Tabla 5.1: Asignación de valores a las variables.

Campo	Variable	Peso
Delimitador	X_1	2 bytes
Longitud	X_2	1 bytes
Tipo de trama	X_3	1 bytes
Dirección MAC64	X_4	8 bytes
Dirección MAC16	X_5	2 bytes
Options	X_6	1 bytes
Numero de sensor	X_7	1 bytes
Checksum	X_8	1 bytes
Location	X_9	8 bytes
GPS	X_{10}	15 bytes
Date	X_{11}	19 bytes SQL

Tabla 5.2: Trama Zigbee + SQL.

Pre	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}
6						40					19

$S_x =$ número de sensores para m_i ,

¹Sobreprocesamiento aplicado a la cabecera de la trama enviada

Cada nodo m_i tendrá 8 sensores.

Por lo tanto $m_i = 65\text{bytes} * S_x * 8\text{bits}$

$$m_i = 4160\text{bits}$$

El número de nodos m_i será igual a 8 transmitiendo la variable temperatura. Aplicando al ecuación 8

$$V_{ch} = \sum_{i=1}^8 [m_i \cdot S_x] * \text{overhead}$$

$$V_{ch} = [(4160) \cdot (8)] * [20\%] \text{bps}$$

$$V_{ch} = (33280) \cdot (1,2) \text{bps}$$

$$V_{ch} = 39,93 \text{Kbps}$$

La velocidad del canal, anteriormente calculado, es el tráfico generado por 8 nodos sensores enviando 8 muestras de temperatura por segundo en forma simultánea al nodo central. El modelo de congestión de los datos es un modelo determinístico donde las mismas entradas producirán invariablemente las mismas salidas, también se debe tener en cuenta que a medida que aumenta la congestión de datos, determinar el tráfico total se puede asociar con sistemas lineales que no contemplan incertidumbre por lo cual el tráfico se puede modelar en forma discreta si es un sensado a una tasa constante. No obstante, la optimización del canal depende de la disminución de la capacidad del tráfico por ello utilizar modelos complicados que incluyan eventos aleatorios o bien probabilidades de bloqueo tal como lo sugiere poisson² es dificultar el dimensionamiento y diseño de red por lo cual se sugiere trabajar con un modelo determinístico.

Tomando como criterio para seleccionar la velocidad de salida o conexión a internet:

$$V_{ch} < \text{velocidad de salida}$$

Las alternativas para la velocidad del enlace o conexión a internet son:

²Es una distribución de probabilidad discreta que expresa, a partir de una frecuencia de ocurrencia media, la probabilidad de que ocurra un determinado número de eventos durante cierto período de tiempo. Concretamente, se especializa en la probabilidad de ocurrencia de sucesos con probabilidades muy pequeñas, o sucesos "raros".

a. UMTS³

Velocidad = 384 Kbps

GoS = 5 %

Latencia = 5ms

Con este tipo de conexión se tendría una ocupación del 10.4 %. Por lo tanto se cumple:

$$Vch < \text{velocidad de salida}$$

b. GPRS⁴

Velocidad = 64Kbps

GoS = 5 %

Latencia = 5ms

Con este tipo de conexión se tendría una ocupación del 62.4 %. Por lo tanto se cumple:

$$Vch < \text{velocidad de salida}$$

c. LTE

Velocidad = 12 Mbps

GoS = 5 %

Latencia = 4ms

Con este tipo de conexión se tendría una ocupación del 0.33 %. Por lo tanto se cumple:

$$Vch < \text{velocidad de salida}$$

De acuerdo a lo anterior se sugiere la tecnología UMTS como alternativa para sensar la variable temperatura tal como lo ilustra la Figura 5.9.

³Sistema universal de telecomunicaciones móviles (*Universal Mobile Telecommunications System o UMTS*) es una de las tecnologías usadas por los móviles de tercera generación, sucesora de GSM, debido a que la tecnología GSM propiamente dicha no podía seguir un camino evolutivo para llegar a brindar servicios considerados de tercera generación.

⁴*General Packet Radio Service (GPRS)* o servicio general de paquetes vía radio creado en la década de los 80 es una extensión del Sistema Global para Comunicaciones Móviles (*Global System for Mobile Communications o GSM*) para la transmisión de datos mediante conmutación de paquetes.

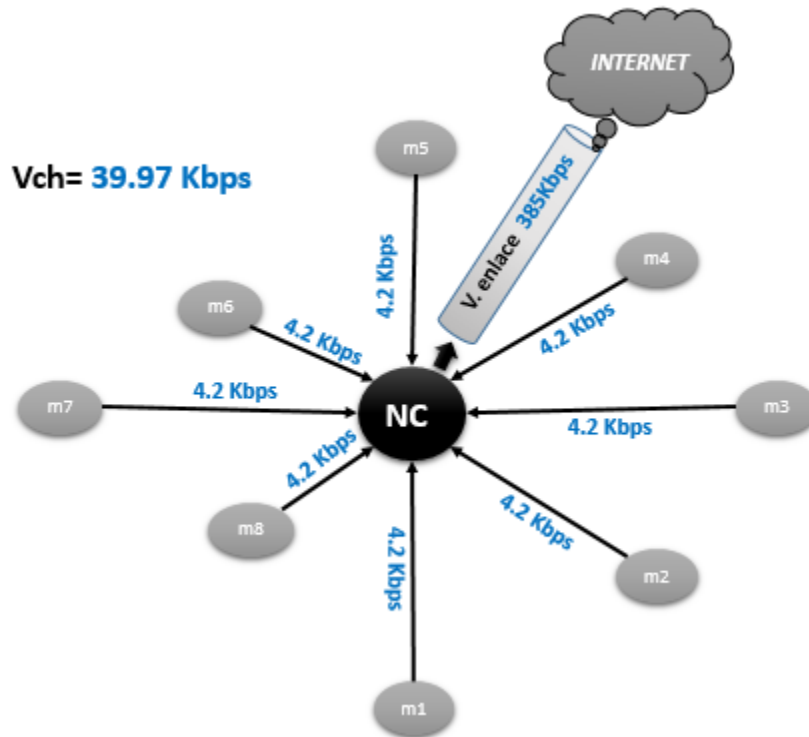


Figura 5.9: Velocidad de enlace para variables lentas (temperatura).

La velocidad del enlace al centro de fusión está determinada por las características de sensado, en donde al escoger la tecnología que se va a utilizar depende de la aplicación que se requiere, es decir, si es una variable de sensado rápido requiere anchos de banda del orden de 12Mbps de lo contrario, si son variables lentas o que no requieren constante monitoreo se puede utilizar anchos de banda con velocidades del orden de 384Kbps. Por lo cual lo diseñado y prometido en capítulo 2 y posteriormente implementado en el capítulo 4 y lo expuesto en este capítulo 5, conforman la arquitectura de red propuesta para la alternativa de solución al problema de escalabilidad y latencia en una WSN como solución de implementación alterna a una arquitectura tipo malla.

Capítulo 6

Análisis, Recomendaciones y Trabajos Futuros

En el primer capítulo se analizaron todas las condiciones de operación de una red de sensores en diferentes escenarios y los aportes realizados por empresas comerciales y academias. Los aportes y análisis se hicieron a través de trabajos y recomendaciones que otros hicieron, adicionalmente se trató el tema de las plataformas “Open source Hardware” en la cual, en el reporte técnico e hicieron algunas comparaciones para determinar cuál es el dispositivo que mejor se adaptaba a las necesidades de este trabajo. En el segundo capítulo se analizó, estudió y diseñó el dispositivo sin puesta en funcionamiento. Por lo cual ahí el aporte es un resultado de tantos que se pueden haber, es decir, el dispositivo se pudo haber diseñado con otros elementos encontrados comercialmente para el caso del ordenador de placa única pudo haberse utilizado en vez de la Raspberry Pi modelo B una Cubieboard, Galileo o bien un pcDuino. Lo mismo aplica para el tema de los microcontroladores se hizo un análisis de por qué los PIC son la mejor opción a la hora de utilizarlos e implementarlos. Por último en este capítulo 2 se hicieron dos diagramas UML para el diseño del software y un modelo entidad relación para el diseño de la base de datos los cuales previamente se les hizo el respectivo análisis. El capítulo 3 se estudió la norma IEEE 802.15.4, Luego se empezó a estudiar los elementos que aportan a la solución y propuesta de una red con topología en estrella a través de un modelo de canal en ambientes urbanos, descripción del link budget y simulación de la red, para finalmente decir cuál va a ser el aporte en condiciones de operación del futuro dispositivo que en ese momento no había sido implementado, por lo tanto se definieron las características de operación del dispositivo en forma teórica y simulada. En el capítulo 4 se implementó el trabajo previamente diseñado y analizado, se desarrollaron algoritmos implementación tanto para el envío de los datos como para la recepción de los mismos. El dispositivo se puso en funcionamiento en pruebas hechas de laboratorio el cual puede ser insertado dentro de la arquitectura de red propuesta en el capítulo 3. En el capítulo 5 el análisis se desarrolló en torno a tres plataformas de conexión a la nube (Xively, ODK y amazon) no obstante la plataforma escogida fue AWS debido a su estabilidad y facilidad de uso, además en esta plataforma tiene

la posibilidad de armar un sistema mucho más complejo y con especificaciones técnicas tal que pueda ofrecer muchas otras aplicaciones. En este sistema se implementó la base de datos y se analizó, para lo cual el dispositivo está funcionando y generando resultados de implementación a través de una arquitectura propuesta.

El dispositivo funciona en condiciones de operación al igual que los algoritmos de implementación y plataforma en la nube a través del diseño e implementación con arquitectura de red en topología estrella de 8 nodos y tramas máximas de 50 bytes con conexión a la nube y con tecnología UMTS. Los cuales aportan al problema de la latencia y escalabilidad, a través de la propuesta de la arquitectura de la red expuesta en el capítulo 3 sin embargo esta propuesta es válida para sensado de variables lentas como por ejemplo temperatura la cual no requiere muchas muestras por minuto para la mayoría de las aplicaciones.

Finalmente en este capítulo se analizan 4 aspectos: diseño, escalabilidad de red, integridad de los datos y rapidez del flujo de información. Sin embargo los diseños que normalmente se implementan utilizan topologías de red en “Mesh o Tree” lo cual implica que la concentración de datos en nodos “Sink” forme cuellos de botellas, afectando la escalabilidad y acarreando retardos en el tráfico de la red [Figura 6.1].

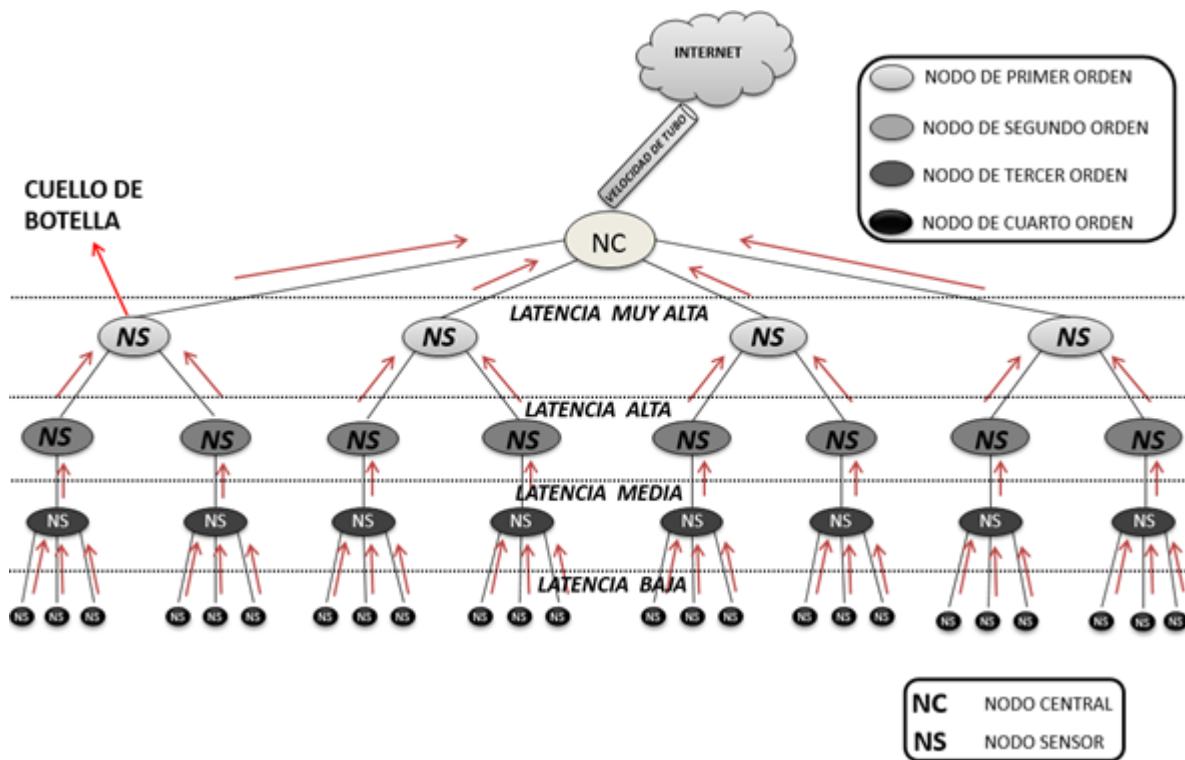


Figura 6.1: Tráfico de datos en una WSN tipo “Mesh”.

6.1. Análisis de Escalabilidad y Latencia en una WSN

Atacar los problemas de escalabilidad y latencia, en forma distribuida, es una alternativa a tener en cuenta. Por ello en este trabajo se diseñó un dispositivo que captura los datos de una red de sensores y los carga en forma directa a un centro de fusión localizado en la nube [Figura 6.2].

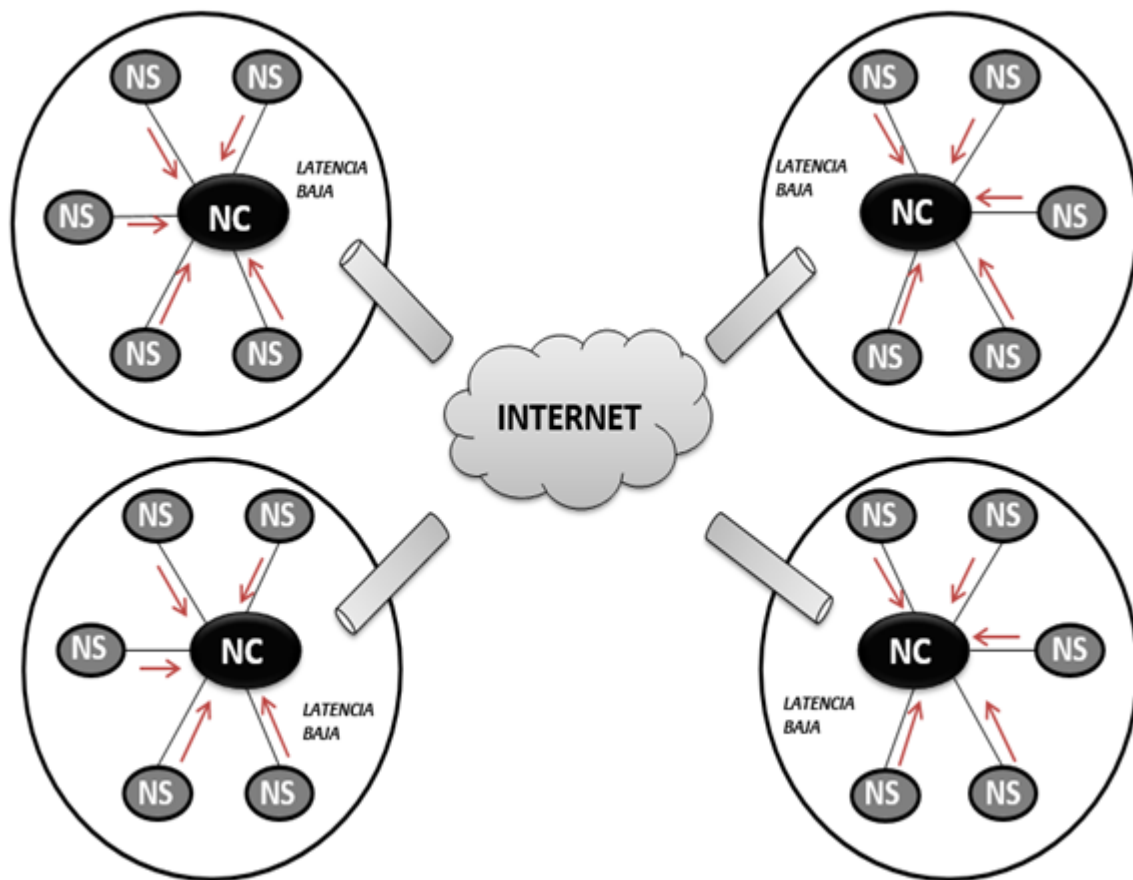


Figura 6.2: Pequeñas redes de sensores tipo Estrella.

Como se evidencia en la figura anterior, el tema de la latencia va asociado al número de saltos de red generado por el recorrido del flujo de información. Adicionalmente la cantidad de tráfico procedente de puntos diferentes acarrea retrasos y colas en ciertos puntos o nodos de la red; por esta razón, entre menos saltos tenga la red, más rápido y óptimo es el desempeño de la misma.

La topología en malla posee muchas ventajas en cuanto a la cobertura y al número de dispositivos que se conectan en forma simultánea aunque sigue siendo una desventaja los problemas de latencia asociados a este tipo de red, por lo cual, los saltos de red generan retrasos y congestión de tramas a lo largo de ella. Para estos casos, si un nodo posee alto tráfico de red pueden suceder los siguientes eventos: bloqueo de red, daños en las tramas de datos, datos erróneos, atenuación de la señal. La gráfica de la Figura 6.3 describe 2 escenarios: uno en el que la información llega a través de 4 estaciones o saltos de red, acarreado escalabilidad baja y latencia alta; mientras que en el escenario número dos, la información llega a través de un solo salto de red por lo cual mejoran los aspectos de escalabilidad alta y latencia baja.

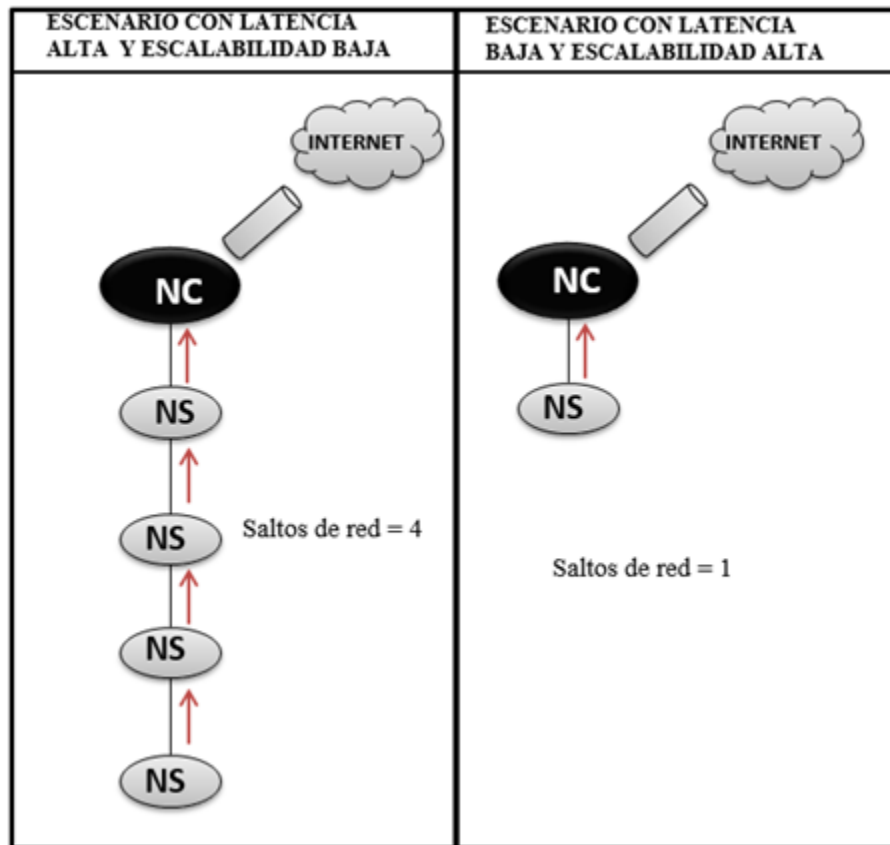


Figura 6.3: Escenarios de escalabilidad.

El dispositivo diseñado e implementado en este trabajo, es utilizado en una red con topología en estrella con acceso a internet de tal forma que los datos que se capturen sean cargados inmediatamente al servidor que se encuentra localizado en internet. Para tratar el tema de la escalabilidad se tienen dos escenarios:

a. En el primer escenario, [Figura 6.4] se tiene un nodo coordinador con 100 nodos sensores conectados enviando información.

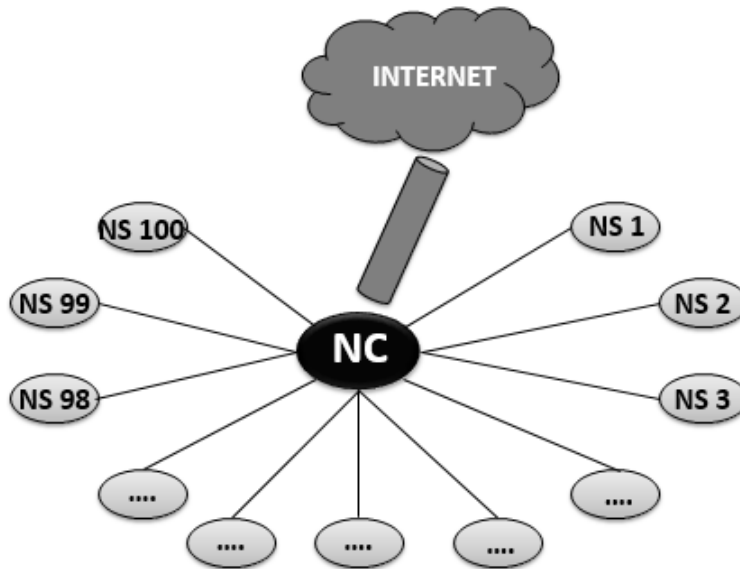


Figura 6.4: Un nodo coordinador con 100 nodos sensores.

b. En el segundo escenario, [Figura 6.5] se tienen 10 nodos coordinadores con 10 nodos sensores.

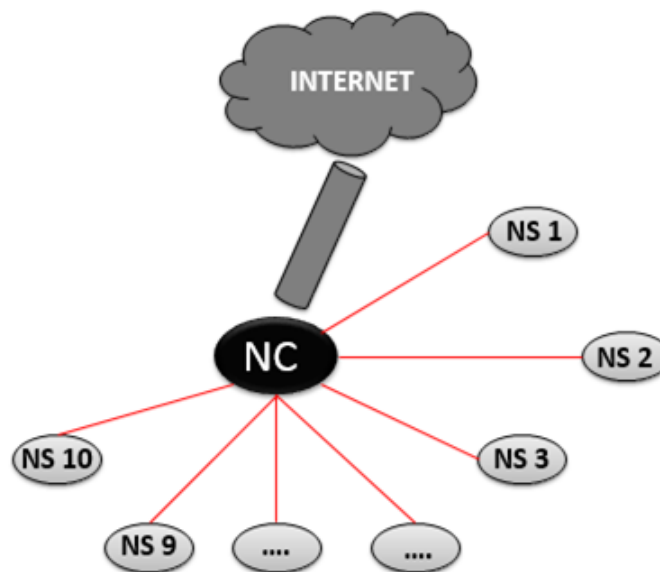


Figura 6.5: Un nodo coordinador con 10 nodos sensores.

Si se comparan los dos casos se analizan los siguientes aspectos para el primero, un nodo coordinador [Figura 6.6]:

- Los costos del enlace de velocidad serán muy significativos puesto que la conexión debe ser de altas velocidades.
- Aumenta la latencia a medida que todos los sensores envían información simultánea
- La velocidad de transmisión debe ser alta.
- La complejidad de implementación aumenta puesto que es una red más robusta.

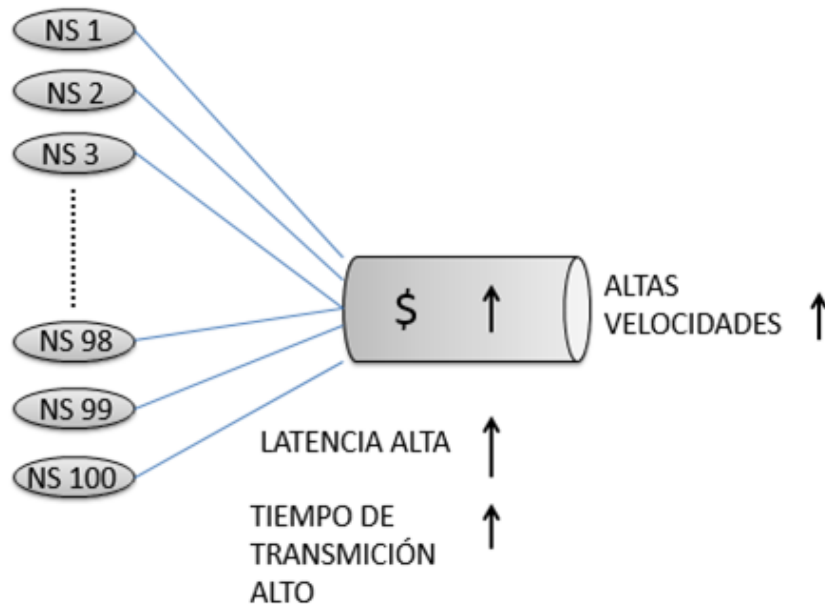


Figura 6.6: Aspectos del primer escenario.

En el segundo escenario lo más importante es conservar la escalabilidad y disminuir la latencia de tal manera que la integridad y seguridad de los datos no se vean en riesgo, para lo cual, según la Figura 6.7 se mejoran aspectos como latencia baja debido a la disminución de saltos de red, velocidades medias de conexión es decir ya no se necesita un ancho de banda muy grande para abarcar las conexiones de red a la nube tal. Sin embargo, esta opción puede ser más costosa de implementar si se tiene en cuenta que la variable a medir no requiere de mayor atención, en cambio si las variables a medir son de precisión y ponen en riesgo un proceso o monitoreo de cuidado, el costo no debe importar.

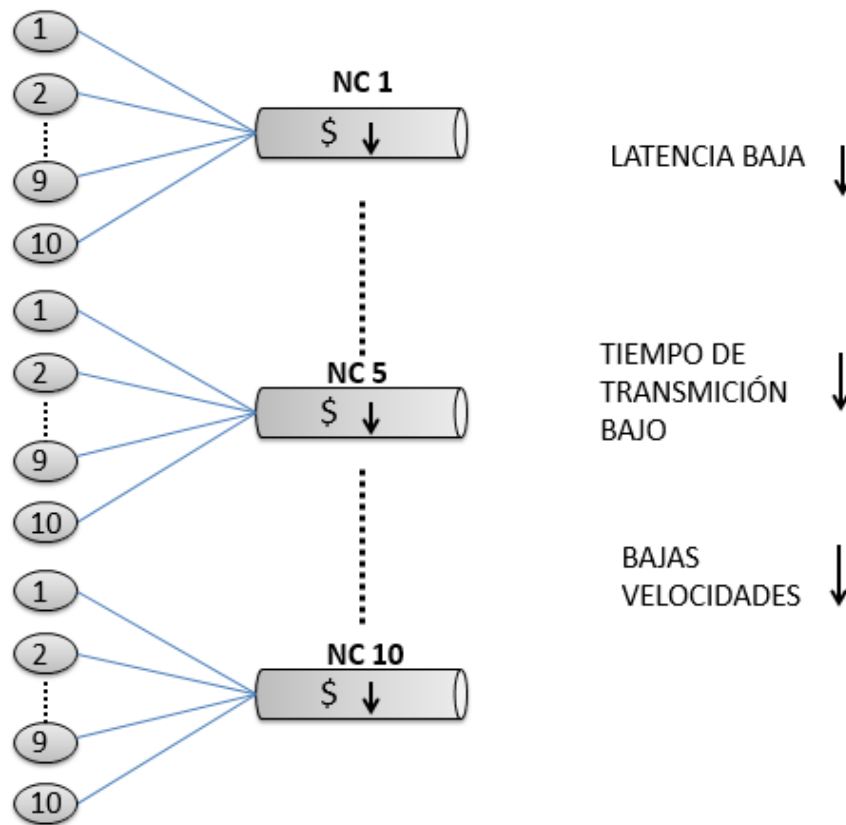


Figura 6.7: Aspectos del segundo escenario.

En el escenario 2, estos aspectos son solo válidos si se requiere una implementación de pequeñas redes de sensores en forma distribuida y sencilla de manejar. Sin embargo, es mejor implementar diez redes de sensores en topología en estrella con “10” nodos sensores que una red de sensores en topología en estrella con “100” nodos sensores si solo si se tiene en cuenta que la prioridad es la integridad de los datos, velocidad de procesamiento, sensado preciso, optimización en el tráfico, rapidez en la entrega de los paquetes, eficiencia energética, latencia baja y escalabilidad.

6.2. Recomendaciones

Después del desarrollo del presente trabajo, se recomienda lo siguiente:

a. En cuanto al trabajo de investigación: Buscar empresas dedicadas al servicio de instalación de redes de sensores para consultar los problemas asociados a la implementación de las mismas al igual que los protocolos de tipo “open” que puedan ser modificados para hacer

estándares de implementación mucho más asequibles y menos costosos.

b. En cuanto al diseño del dispositivo: El diseño de hardware utiliza elementos de bajo costo por lo cual sus características son muy limitadas para ello se recomienda utilizar dispositivos para futuros diseños tales como: microcontroladores DSP, Ordenadores de placa reducido de dos o cuatro núcleos, módulos inalámbricos de mayor potencia de transmisión y velocidad. Para la parte de software se debe diseñar a partir de lenguajes de programación mucho más complejos que ofrezcan alternativas más óptimas de procesamiento de información y almacenamiento de datos.

c. En cuanto al dimensionamiento de la red: Realizar pruebas con protocolos cerrados o con uso de licencia, de tal manera que se evidencien si hay o no alguna ventaja en relación al Zigbee y luego sacar tablas de comparaciones entre una tecnología y otra. También, se recomienda tener en cuenta los problemas asociados a la conectividad Wi Fi disminuyéndolos.

d. En cuanto la implementación: Utilizar elemento de bajo consumo energético al igual que dispositivos que soporten largas horas de trabajo con alto tráfico de red.

e. En cuanto al centro de fusión: Evaluar los ítems de seguridad y disponibilidad de la información de las plataformas especializadas en guardar datos de WSN diferente a la que se utilizó en este trabajo.

6.3. Trabajos Futuros

Los problemas asociados a redes de sensores siguen siendo temas abiertos por lo cual, los trabajos futuros deben enfocarse en:

- La optimización de recursos energéticos
- Implementación de protocolos de comunicación abiertos
- La realización pruebas con recursos tecnológicos con características más especializadas.
- La generación de algoritmos eficientes y óptimos para la captura de datos.
- La valoración la eficiencia en el procesamiento de la información.
- La búsqueda de alternativas para la seguridad de hardware en condiciones de operación.

- Seguridad para el acceso de los datos al igual que su disponibilidad.
- Sincronización de los datos para sensado con muestras simultáneas.

Para ello se propone que el dispositivo tenga un sistema de consumo energético utilizando algún tipo de energía renovable o fotovoltaica, al igual que buscar un estándar o protocolo diferente al Zigbee que ofrezca arquitectura IPv6. También, el dispositivo se puede implementar ordenadores de placa reducida de dos y cuatro núcleos para aplicaciones más complejas. En cuanto a la seguridad de hardware debe diseñarse una carcasa para introducir el dispositivo y pueda resistir condiciones ambientales externas. La seguridad en la base de datos que almacena la información debe crearse con algoritmos mucho más seguro y que no afecten la integridad y disponibilidad de la información. Finalmente, el dispositivo debe ser capaz de procesar información simultánea en tiempo real de tal manera que el tema de sincronización de datos simultáneos sea una opción más para implementarse.

Capítulo 7

Conclusiones

En el marco del internet de las cosas, la arquitectura de redes de sensores posee grandes cantidades de sensores enviando información a centros de datos los cuales procesan la información y posteriormente guardarla. La mayoría de estas topologías de red suelen ser en malla, por tal razón, la concentración de datos en puntos específicos de la malla suele formar cuellos de botella. Cuando esta eventualidad se presenta, la red no podrá soportar el envío masificado de datos lo cual acarrea problemas de escalabilidad y latencia debido al tipo de arquitectura en la red. Como posible alternativa para que esto no suceda, se debe implementar un sistema distribuido en pequeñas redes de sensores enviando la información a un centro de fusión ubicado en un servidor en internet, lo cual ataca directamente los problemas de escalabilidad y latencia de una arquitectura tipo malla. Además esta red distribuida de nodos coordinadores al recibir datos de unos pocos nodos sensores, envía la información a una base de datos disminuyendo de muchos a “uno”, los saltos que en una red “Mesh” podrían ser de 4 a 10 saltos. Por su parte las plataformas tecnológicas que existen actualmente en el mercado permiten un diseño que se ajusta a las necesidades para suplir los problemas anteriormente mencionados tales como: costos, información de los dispositivos, características técnicas, variedad de los dispositivos, alcance, cobertura de red, protocolos híbridos de comunicación y por supuesto todas aquellas herramientas que un diseñador puede encontrar. Entre estas plataformas tecnológicas se tienen: módulos inalámbricos de comunicación a nivel de capa MAC (*Xbee*), Ordenadores de placa reducida (*Raspberry Pi*, *Galileo*, *Pcduino*), protocolo de comunicación de bajo consumo energético (*Zigbee*), herramientas de simulación de software (*Opnet IT*, *Scratch*, *Labview*, *Proteus*), plataformas open hardware (*Arduino*), plataformas open software (*MySQL*, *Php*), ambientes integrados de desarrollo (*Mplab de Microchip*, *Python*, *Xctu de Digi*), servidores gratuitos en la nube (*Amazon Web services*), dispositivos electrónicos programables (*PIC Microchip Atmel*). Todas estas herramientas y plataformas tecnológicas son asequibles por costos, disponibilidad y documentación y sirven para diseñar prototipos de sistemas de redes inalámbricas de sensores en forma distribuida conjugando lenguajes de alto nivel con dispositivos a nivel de hardware.

El diseño del dispositivo propone que el manejo de los tiempos de latencia sean muy bajo al igual que la red sea escalable a medida que las necesidades de la aplicación lo requieran por lo que se seleccionaron los siguientes recursos tecnológicos que conformaron el dispositivo final que captura los datos de una WSN y los carga en forma directa a un centro de fusión: Raspberry Pi (por su bajo costo \$35, procesador e interfaz Ethernet), Módulos Xbee serie 2 (Por su implementación en la norma IEEE 802.15.4 de bajo consumo energético, compatibilidad con otros dispositivos que manejen el protocolo Zigbee, uso de la banda 2.4Ghz y bajo costo), Microcontrolador PIC (para que realice una sola tarea: la conversión a análogo digital de la información enviada por un sensor). Entre tanto el diseño de software debe contener los siguientes aspectos: identificar los actores del sistema para una red distribuida de un nodo coordinador y unos pocos nodos sensores asociados al igual que normalizar la base de datos en donde se encuentra la información de dichos nodos, para ello se utilizaron casos de uso para la descripción del sistema y un modelo entidad relación para el diseño de la base de datos. El sistema es totalmente autónomo, la información y funcionamiento como tal se encuentra almacenada en la tarjeta SD (en donde se encuentra sistema operativo y herramientas de software) de la Raspberry Pi lo cual facilita la implementación de un sistema a escala (es posible implementar muchos de estos en una zona geográfica determinada). Además se implementó con los recursos tecnológicos más favorables en precios del mercado de tal manera que su implementación no manejará un presupuesto alto sin que se vean afectadas las características que un sistema de estos requiere (optimización y velocidad en el procesamiento de la información), hacer este dispositivo con otros elementos incrementaría en el triple los costos que este se llevó. Este diseño, a diferencia de los otros el mercado que suelen ser robustos, es sencillo, simple y de fácil comprensión puesto que las conexiones físicas mínimas y sus elementos son de bajo consumo energético también tiene la posibilidad de interactuar con otros dispositivos de diferentes marcas.

Una red Zigbee con topología estrella, cuyo nodo coordinador recibe tramas provenientes de ocho nodos router de 14 paquetes de tamaño de 50 bytes que ocupa un 49% de la capacidad del enlace si su velocidad de transmisión es de 170Kbps, puede cubrir un área geográfica de 50x50 metros ofreciendo una red escalable y baja en latencia. Estas condiciones tienen una cobertura de 2.5Km² con 112 sensores en toda el área enviando información cada segundo y la escalabilidad de la red depende de la cantidad de información que se requiere por minuto. Si se aumentan los intervalos de envío de paquetes se pueden instalar más sensores incluso más nodos, aún con todo esto el canal puede suplir más necesidades si se requieren en un momento determinado; reconfigurando la red para que se adapte a las nuevas condiciones de trabajo, es decir, Se pueden instalar nodos en modo “*End device*” los cuales no exigen mayores recursos de red y realizan funciones más específicas y tienen la capacidad del modo “*Sleep*” que disminuye consumo energético. Se puede hablar de una red escalable de 10 nodos router, 10 nodos “*end device*”, un coordinador y 280 sensores en un área de hasta 15.6Km², utilizando la mayor capacidad del “*Link Budget*”. La topología en estrella permite que los datos lleguen en un solo salto de red por lo cual es válida para sensado rápido

sin embargo se aclara que el sistema al utilizar la banda 2.4Ghz adquiere todos los problemas de interconectividad generados por el Wi Fi.

La implementación del dispositivo se hace a través de elementos de hardware, software y diseño los cuales, cuando se conjugan forman un sistema que captura los datos de un nodo sensor conformado por tres elementos un sensor, un microcontrolador (PIC16F1937) que procesa la señal de voltaje y envía la información a un módulo Xbee (router) encargado de enviar la trama en forma inalámbrica al otro bloque del sistema llamado nodo coordinador, conformado por un Módulo Xbee (coordinador) recibiendo y posteriormente enviando la información de la trama recibida a una Raspberry pi, la cual procesa la información recibida a través del sistema operativo “*Raspbian*” en uno de sus archivos “*ttyAMA0*” y por medio de un algoritmo en php se conecta a la base de datos local “*BD_sensores1*”. Este sistema propone la integración de configuraciones que no pueden ser alteradas tales como: velocidad de los dispositivos normalizados “9600Bps 8N1”, la alimentación de los dispositivos está entre 3,3V y 5V y la sincronización entre la trama recibida y el algoritmo que recibe esta información esta última anotación se deja planteado que para solucionar este tema de sincronización debe ir asociado a la la creación de un algoritmo de seguridad al momento de ingresar la información en la base de datos, verifique que la información ingresada sea la correcta asegurando la integridad de los datos. No obstante, la modificación o alteración de cualquiera de los elementos alteraría el proceso de guarda de datos pero posee a su favor tener un alto nivel de abstracción en donde se puede interpretar con facilidad el funcionamiento de todo el sistema. Implementar este sistema posee ventajas de tipo económico porque reduce costos ya que los recursos tecnológicos utilizados son los más favorables del mercado porque permite hacer mantenimiento correctivo y preventivo en cualquier momento, si solo si, alguno de sus bloques se encuentra en funcionamiento deficiente o deja de funcionar, la información de diseño es muy asequible en medios como internet, el consumo energético es muy reducido incluso puede instalarse un sistema autónomo de baterías recargables con minipaneles solares, también se pueden implementar seguridad de operación y procesos, es un sistema distribuido escalable que puede realizar múltiples funciones (no solo recoger datos) y utiliza elementos y plataformas “*Open*” los cual disminuye costos de licencia. El script php que se ejecuta a través de un algoritmo bash es lo más cercano al núcleo del sistema operativo lo que permite que sea un sistema de recolección de datos óptimo y eficiente lo cual disminuye tiempos de latencia y asegura escalabilidad. En general la implementación para trabajos futuros puede utilizarse para espacios inteligentes o como recolector de variables en zonas de alto riesgo.

Las plataformas de almacenamiento de datos de sensores localizadas en la nube ofrecen ventajas de integridad, disponibilidad y seguridad de la información, por cuanto sus características permiten a cualquier usuario acceder a la información almacenada en cualquier instante de tiempo. La plataforma que se utiliza para guardar los datos en la nube ofrece los servicios anteriormente descritos y para acceder a esta plataforma, el dispositivo realizado posee una “velocidad de canal colector” generado por todos los nodos sensores conectados

al nodo coordinador para ello el ancho de banda de conexión a internet debe ser de una velocidad mayor tal que la tasa de transmisión de cada uno de los nodos, junto con las cabeceras de información y señalización del canal colector sea lo suficiente para que se transmitan los datos a una latencia aceptable en la aplicación de uso. Por tanto la conexión seleccionada es de tecnología UMTS que para el caso de la variable temperatura su conexión puede llegar hasta 385Kbps, si se tiene en cuenta que el tráfico generado por toda la red propuesta es de 39.97Kbps, incluido el sobreprosesamiento que solo un 10.3% de la capacidad del canal. El modelo de congestión de datos es de tipo determinístico y el tráfico se puede modelar en forma discreta, para no complicar el diseño con otros modelos que sugieran la inclusión de eventos aleatorios o sucesos inesperados. Una vez capturados y guardados los datos en la nube se puede hacer inteligencia y toma de decisiones, que permiten aportar información a los centros de monitoreo ambiental. Este dispositivo, debido a su configuración de hardware y topología de red, facilita la organización de la información porque permite identificar en el menor tiempo posible la falla de algún nodo conectado al nodo principal.

Bibliografía

- [1] A. Domingo, B. Bellalta, M. Palacin, M. Oliver y E. Almirall, «Public Open Sensor Data: Revolutionizing Smart Cities,» *Technology and Society Magazine, IEEE*, vol. 32, nº 4, pp. 50-56, winter 2013.
- [2] L. Filipponi, A. Vitaletti, G. Landi, M. Vincenzo, G. Laura y P. Pucci, «Smart City: An Event Driven Architecture for Monitoring Public Spaces with,» de *Fourth International Conference on Sensor Technologies and Applications*, 2010.
- [3] G. Cardone, P. Bellavista, A. Corradi y L. Foschini, «EFFECTIVE COLLABORATIVE MONITORING IN SMART CITIES:,» *Bologna*, 2013.
- [4] M. Fazio, M. Paone, A. Puliafito y M. Villari, «Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing,» *Messina*, 2012.
- [5] «Internet de Las Cosas,» 2012. [En línea]. Disponible: <http://www.internetdelascosas.cl/que-es/>. [Último acceso: 9 05 2014].
- [6] P. Manchego, «Redes de Datos de Sensores. Metodología para la Recolección, Administración y Presentación de Datos.,» *Revista SG*, pp. 60 - 65, 2007.
- [7] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. Glaser y K. Pister, «OpenWSN: a standards-based low-power wireless development environment,» *Transactions on Emerging Telecommunications Technologies*, vol. 23, nº 5, pp. 480-493, 2012.
- [8] «ceeiaron,» 2014. [En línea]. Disponible: <http://www.ceeiaron.es/empresa-detalle/93/libelium/>. [Último acceso: 15 11 2014].
- [9] G. Diaz, «Redes de Sensores en la Prevención de Desastres,» *Panamá*, 2013.
- [10] M. Sudheer, *Wireless Sensor Network for Disaster Monitoring, Wireless Sensor Networks: Application-Centric Design*, Yen Kheng Tan, 2010.
- [11] «Using Wireless Sensor Networks in the Sensor Web for Flood Monitoring,» de *10th International ISCRAM Conference, Brazil*, 2013.

- [12] M. E. Gordon Pico y B. S. Vásquez Castro, «Análisis de Desempeño de Redes de Sensores Inalámbricas Aplicadas a Monitorización Volcánica,» Quito, 2012.
- [13] L. N. Mosquera Gómez, «Red de Sensores Para Monitorear Bosque Húmedo,» Medellín, 2014.
- [14] U. C. f. A. Research, «Guía de Referencia para Sistemas de Alerta Temprana de Crecidas Repentinas,» 2012.
- [15] W. M. Peng Jia, «Wireless Remote Monitor and Control System Based on ZigBee and Web,» Intitute of Beijing, Beijing, 2013.
- [16] I. J. Zaradnik, «ElectroComponentes,» 03 02 2014. [En línea]. Disponible: http://www.electrocomponentes.com/educacion/download/SASE_2014_DIGI_Tutorial_Modulos_XBEE.pdf. [Último acceso: 04 11 2014].
- [17] «dataweek,» 06 08 2008. [En línea]. Disponible: <http://www.dataweek.co.za/news.aspx?pklnwsid=29778>. [Último acceso: 08 01 2014].
- [18] «informaticayrsec,» 12 05 2011. [En línea]. Disponible: <http://informaticaceayrsec.blogspot.com/2011/05/12-sistema-informatico.html>. [Último acceso: 23 10 2014].
- [19] «developer-blog,» 09 2013. [En línea]. Disponible: <http://developer-blog.net/hardware/raspberry-pi-gpio-schnittstelle-teil-1/>. [Último acceso: 08 02 2015].
- [20] AlvaroUnal, 12 2011. [En línea]. Disponible: <http://alvarounal.blogspot.com/2011/12/modulos-xtbee-parte-6-configuraciony.html>. [Último acceso: 12 11 2014].
- [21] «webdelcire,» 02 2014. [En línea]. Disponible: <http://webdelcire.com/wordpress/archives/1714>. [Último acceso: 05 11 2014].
- [22] Varios, «Word Press,» [En línea]. Disponible: <http://jonybattery7.wordpress.com/>. [Último acceso: 1 10 2014].
- [23] Microchip, «Hoja de Datos Microcontrolador PIC16F1937,» Arizona, 2008.
- [24] «openpicus,» 2013. [En línea]. Disponible: <http://community.openpicus.com/forum/flyport-modules-hardware/3845/flashing-bootloader-pickit3>. [Último acceso: 23 11 2014].
- [25] P. 1. IEEE Standard 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks, Madrid : Gupo 802.15, 2006.

- [26] J. P. Dignani, «Análisis de protocolo Zigbee,» Universidad Nacional de La Plata, La plata , 2011.
- [27] G. A. Alvarez, «El Standard ZigBee, Universidad Nacional de Trujillo,» [En línea]. Disponible: <http://www.scribd.com/doc/19049146/El-Standard-ZigBee>. [Último acceso: 29 12 2014].
- [28] O. Musikanon y W. Chongburee, «ZigBee Propagations and Performance Analysis in Last Mile Network,» International Journal of Innovation, Management and Technology, Bangkok, 2012.
- [29] C. A. Castillo Luzón , «Implementación de un prototipo de red de sensores inalámbricos para invernaderos,» Escuela politécnica nacional , Quito, 2007.
- [30] P. A. Gómez Puerta, G. A. Posada Pérez y M. A. Vallejo Velásquez, «Evaluación del desempeño del protocolo de enrutamiento AODV para diferentes escenarios de redes de sensores inalámbricos,» Universidad del Norte , Barranquilla , 2013.
- [31] Á. Armendáriz Mateos y F. J. Falcone Lanás, «Análisis de la dependencia topológica de una red de sensores Zigbee en relación con el canal de radioprogramación para entornos interiores,» Universidad pública de Navarra, Pamplona, 2011.
- [32] I. NTS, «Norma ISO 27001,» ICONTEC, Bogotá, 2014.
- [33] Etherios, «Your Internet of Anything Platform,» 2014. [En línea]. Disponible: <http://www.etherios.com/products/devicecloud/>. [Último acceso: 30 12 2014].
- [34] Etherios, «Device Cloud,» 1 09 2008. [En línea]. Disponible: <http://www.etherios.com/products/devicecloud/>. [Último acceso: 3 11 2014].
- [35] «internet de las cosas,» 07 2013. [En línea]. Disponible: <http://www.internetdelascosas.cl/wp-content/uploads/2013/07/xively.jpg>. [Último acceso: 30 12 2014].
- [36] LogMeIn, «Xively,» 2014. [En línea]. Disponible: https://xively.com/whats_xively/. [Último acceso: 30 12 2014].
- [37] P. Swabey, «Information Age,» 3 05 2013. [En línea]. Disponible: <http://www.information-age.com/technology/mobile-and-networking/123457051/were-the-amazon-web-services-of-the-internet-of-things-says-xively>. [Último acceso: 1 11 2014].
- [38] «ODK Sensors Framework,» 12 04 2013. [En línea]. Disponible: <https://play.google.com/store/apps/details?id=org.opendatakit.sensors>. [Último acceso: 30 12 2014].

- [39] ODK, «Open Data Kit,» 13 03 2013. [En línea]. Disponible: <https://opendatakit.org/use/sensors/>. [Último acceso: 10 12 2014].
- [40] «bcnbit consultoria y sistemas,» 2013. [En línea]. Disponible: <http://www.bcnbit.com/5-razones-para-utilizar-el-cloud-de-amazon/>. [Último acceso: 30 12 2014].
- [41] Varios, «Xtaka,» 8 03 2014. [En línea]. Disponible: <http://www.xataka.com/ordenadores/raspberry-pi-el-ordenador-de-los-25-dolares-inicialmente-en-ebay>.
- [42] C. O. t. A. Poverty, «Carbon Offset to Alleviate Poverty,» 06 10 2013. [En línea]. Disponible: <http://cotap.org/>.
- [43] T. Zimmermann, H. Wirtz, O. Punal y K. Wehrle, «Analyzing Metropolitan-Area Networking within Public Transportation Systems for Smart City Applications,» de New Technologies, Mobility and Security (NTMS), 2014 6th International Conference on, 2014.
- [44] M. E. Pardo, E. s. Guillermo y D. Martínez, «Una implementación de un sistema de control domótico basada en servicios web,» Buenos Aires, s.f.
- [45] D. L. GALGODA LIYANAGE, «BACHELOR OF ENGINEERING DEGREE WITH HONOURS IN DIGITAL COMMUNICATIONS AND ELECTRONICS,» University of Hertfordshire, Londres, 2010.
- [46] W. Brunette, R. Sodt, R. Chaudhri, M. Goel, M. Falcone, J. Van Orden y G. Borriello, «Open Data Kit Sensors: A Sensor Integration Framework for Android at the Application-Level,» Washington, 2012.
- [47] A. Bonganay, J. Magno, A. Marcellana, J. Morante y N. Perez, «Automated electric meter reading and monitoring system using ZigBee-integrated raspberry Pi single board computer via Modbus,» de Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference on, 2014.
- [48] [1] U. I. d. Telecomunicaciones, «Unión Internacional de Telecomunicaciones,» [En línea]. Available: <https://www.itu.int/rec/T-REC-Y.2069-201207-I/en>. [Último acceso: 23 06 2015].