



UNIVERSIDAD DE CARABOBO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA  
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



**DESARROLLO DE UN PROTOTIPO DE RED DE SENSORES  
INTELIGENTES INALÁMBRICOS BASADO EN EL ESTÁNDAR IEEE  
802.15.4 PARA USO EN AMBIENTES INDUSTRIALES**

**Tutor:**  
**Ángel Villegas**

**Autores:**  
**Dany E. Nowak G.**  
**Ricardo A. Estévez R.**

**Bárbula, Julio de 2013**



UNIVERSIDAD DE CARABOBO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA  
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



DESARROLLO DE UN PROTOTIPO DE RED DE SENSORES  
INTELIGENTES INALÁMBRICOS BASADO EN EL ESTÁNDAR IEEE  
802.15.4 PARA USO EN AMBIENTES INDUSTRIALES

TRABAJO ESPECIAL DE GRADO PRESENTADO ANTE LA ILUSTRE  
UNIVERSIDAD DE CARABOBO PARA OPTAR AL TÍTULO DE  
INGENIERO ELECTRICISTA

Tutor:  
Ángel Villegas

Autores:  
Dany E. Nowak G.  
Ricardo A. Estévez R.

Bárbula, Julio de 2013



UNIVERSIDAD DE CARABOBO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA ELÉCTRICA  
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA



## CERTIFICADO DE APROBACIÓN

Los abajo firmantes miembros del jurado asignado para evaluar el trabajo especial de grado titulado **“Desarrollo de un Prototipo de Red de Sensores Inteligentes Inalámbricos Basado en el Estándar IEEE 802.15.4 para Uso en Ambientes Industriales”**, realizado por los bachilleres: Dany E. Nowak G. y Ricardo A. Estévez R., cédulas de identidad: V-18.168.268 y V-19.481.900 respectivamente, hacemos constar que hemos revisado y aprobado dicho trabajo.

Prof. Ángel Villegas  
TUTOR

Prof. Demetrio Rey  
JURADO

Prof. Luis Llave  
JURADO

Bárbula, Julio de 2013





# Dedicatorias

*A mi padre Dany Nowak, mi madre Carmen de Nowak, mi hermana Lía Nowak y mi tío Juan Manuel López quienes han sido, y serán siempre, las personas más importantes de mi vida.*

*Dany Nowak*



*A mi madre, a mi hermano y a Dios.  
Ricardo Estévez*

# Agradecimientos

## Dany Nowak

- Primeramente quiero agradecer a mis padres, quienes han sido y serán siempre, mis más grandes amigos, mi gran inspiración, mi gran apoyo y mi fuerza para seguir adelante. A ellos, debo mi vida, el más grande de todos los regalos. Hoy les agradezco, desde el fondo de mi corazón, todo el esfuerzo y el sacrificio que han dedicado, durante estos largos veintiséis años, para que yo pudiera llegar a ser todo lo que soy hoy. No existen palabras que expresen la dicha que siento al tenerlos como padres y agradezco a la vida, el permitirme retribuirles un poco de la inmensa felicidad, con la que han llenado cada uno de mis días, con este gran y valioso logro.
- A mi hermana, Lic. Lía Daniela Nowak Giordanelli, por siempre ser mi gran amiga y brindarme su apoyo incondicional, sin importar hora, lugar, ni motivo; siempre ha sido la mano amiga con la que he podido contar. Con ella he compartido, los momentos más felices de mi infancia, momentos que recordaré por siempre. A ella quiero agradecerle por impulsarme siempre a seguir adelante, por siempre tener las palabras correctas, por siempre tener ese trato especial y ese cariño sin límites de su gran corazón. Hoy me siento el hermano más afortunado del mundo por tenerla a mi lado y me llena de orgullo cada día, con cada uno de sus grandes logros.
- A mi tío Juan Manuel López (Q.E.P.D.), por enseñarme que, sin importar las circunstancias, el rendirse nunca es una opción; por enseñarme a sonreír a la vida a pesar de todo y luchar siempre hasta el final. En él, siempre encontré un gran amigo, el cual siempre recordaré con gran cariño y aprecio. Donde quiera que esté en este momento, espero que sepa, lo importante que fue para mí. Agradezco a la vida, por haberlo hecho parte de mi familia y brindarme la oportunidad de ayudarlo en todo lo que estuvo a mi alcance y más, para hacer de sus últimos años de vida, los más felices posible.
- A mi compañero de tesis y gran amigo, por su gran esfuerzo durante este trabajo de grado. Es motivo de orgullo para mí, el haber logrado esta gran meta junto a su persona. Siempre tendrá mi respeto y confianza profesional por su gran y admirable nivel académico. Es, sin lugar a dudas, una de las personas más brillantes que conozco. Estoy seguro que será un gran profesional y siempre podrá contar con mi amistad y mi apoyo incondicional.
- A mi tía, Mg. Belkys Giordanelli, por su gran ayuda y apoyo con este trabajo de grado. Le agradezco por siempre tener ese trato especial y por siempre brindarme su mano

amiga en todo momento. Siempre será un orgullo para mi, tener la dicha de ser su sobrino.

- Agradezco a mis tíos Efsio Giordanelli y Tamara de Giordanelli por brindarme su gran apoyo durante toda mi carrera. A pesar de la distancia siempre están en mi mente y mi corazón.
- A mi gran amigo, Manuel Esteban Afanador Martínez, por brindarme siempre su valiosa amistad y por siempre estar, incondicionalmente, en los momentos críticos y difíciles de mi vida; en los momentos que demuestran la verdadera amistad. Hoy lo considero mucho más que sólo un amigo, lo considero mi hermano.
- A mis grandes amigos José Estrada y Francisco García por brindarme su ayuda en todo lo que estuvo a su alcance además de su gran amistad y apoyo. A ellos debo gran parte de este gran logro. Ambos me han brindado su valioso apoyo durante gran parte de mi carrera y siempre tendrán mi gran agradecimiento y amistad.
- Al Ing. Enrique Alejandro Estévez por brindarnos su valiosa ayuda durante este trabajo de grado.
- A Rosario Del Carmen Rodríguez Díaz por su gran apoyo durante todo este trabajo de grado. Quiero agradecerle, con mucho aprecio y cariño, por todo su apoyo y colaboración en el logro de esta meta y por su gran cordialidad, amabilidad, hospitalidad y atención.
- Al tutor de esta tesis, Ing. Ángel Villegas, por su trabajo y dedicación permanente y continua al trabajo de investigación, así como sus sugerencias y observaciones, siempre inteligentes y oportunas.
- A Gabriela Acuña, por brindarme su apoyo en los momentos más difíciles de mi carrera.

## Ricardo Estévez

- A Dios por haberme guiado a lo largo de mi carrera, por ser mi fortaleza en tiempos de debilidades y por llenarme el camino de aprendizajes, experiencias y felicidad.
- A mi madre, por el gran amor y devoción que tienes a tus hijos, por el apoyo ilimitado e incondicional que siempre me has dado, por tener la fortaleza de salir adelante sin importar los obstáculos, por haberme formado como un hombre de bien. No hay palabras para agradecerte mamá.
- A mi hermano que nunca titubeo en ayudarme en todo momento. Gracias por tu tiempo, tus consejos y tu apoyo incondicional sin el cual este trabajo no hubiese sido posible.
- A mi familia que me han apoyado en este proyecto de vida como es encontrar en la Ingeniería Eléctrica mi futuro y mi pasión, además de entregarme los valores que han sido mi base para afrontar la vida con dignidad.
- A tía Maite quien, sin su enorme cariño y apoyo desinteresado e incondicional, esta tesis no hubiese llegado a su culmen.
- A mi compañero de tesis, con quien compartí tiempos de grandes dificultades pero finalmente todo resulto bien gracias al trabajo constante.
- A mis maestros que con sus conocimientos y ayuda oportuna y desinteresada contribuyeron a la finalización de este proyecto.
- A mi director y tutor de tesis, su valioso conocimiento, sus sugerencias, su colaboración y guía durante la elaboración de este proyecto.
- A mis amigos que siempre estuvieron a mi lado para ayudarme, escucharme, aconsejarme y en ocasiones guiarme.
- Y agradezco a todas aquellas personas que no he mencionado pero que de una u otra manera formaron parte del desarrollo de este proyecto, brindando su amistad, cariño y ayuda emocional.

Gracias a todos por estar siempre a mi lado.

# Resumen

## DESARROLLO DE UN PROTOTIPO DE RED DE SENSORES INTELIGENTES INALÁMBRICOS BASADO EN EL ESTÁNDAR IEEE 802.15.4 PARA USO EN AMBIENTES INDUSTRIALES.

El presente trabajo de grado tiene como objetivo el desarrollo de un prototipo de red de sensores inteligentes e inalámbricos basada en el estándar de comunicaciones IEEE 802.15.4. La red consta de 4 dispositivos: 3 sensores (Dos de temperatura y uno de presión) y un nodo principal, siendo este último el que permite el enlace entre la red y un computador. Dichos dispositivos, haciendo uso de módulos de comunicación inalámbrica MRF24J40MA, se enlazan bajo un conjunto de protocolos que conforman 4 capas entre sí, siendo estas: la capa Física (PHY), La capa de Control de Acceso al Medio (MAC), la capa de Red y la de Aplicación. Las capas inferiores, conformadas por las capas PHY y MAC, se especifican en el estándar IEEE 802.15.4 y se encargan de la sincronización y la asociación de nodos en la red; mientras que las superiores, conformadas por la capa de Red y la de Aplicación, quienes se encargan del enrutamiento de paquetes y brindar el soporte para protocolos superiores, se soportan en ZigBee.

La red funciona bajo la topología Mesh, la cual le permite a cada nodo comunicarse con cualquier otro y establecer rutas de manera inteligente.

Se desarrolló una interfaz en Visual Studio 2010 para reflejar y ajustar parámetros de la red además de los valores medidos por los anteriormente mencionados sensores.

*Palabras Claves: Red de Sensores, Red Inteligente, IEEE 802.15.4, ZigBee.*

**Tutor:**  
Ing. Ángel Villegas

**Autores:**  
Br. Dany E. Nowak G.  
Br. Ricardo A. Estévez R.

# Índice general

Portada	I
Dedicatorias	III
Agradecimientos	VI
Resumen	IX
Índice General	XV
Índice de Figuras	XX
Índice de Cuadros	XXII
Introducción	XXIII
<b>1. El Problema</b>	<b>1</b>
1.1. Planteamiento del Problema . . . . .	1
1.2. Justificación de la Investigación . . . . .	2
1.3. Los Objetivos de la Investigación . . . . .	3
1.3.1. Objetivo General . . . . .	3
1.3.2. Objetivos Específicos . . . . .	3
1.4. Alcance y Delimitación . . . . .	3
<b>2. El Marco Teórico</b>	<b>6</b>
2.1. Antecedentes . . . . .	6
2.2. Bases Teóricas . . . . .	8
2.2.1. Modelo OSI . . . . .	8
2.2.1.1. Capa Física . . . . .	8
2.2.1.2. Capa de Enlace de Datos . . . . .	10
2.2.1.3. Capa de Red . . . . .	10
2.2.1.4. Capa de Transporte . . . . .	11
2.2.1.5. Capa de Sesión . . . . .	12
2.2.1.6. Capa de Presentación . . . . .	12
2.2.1.7. Capa de Aplicación . . . . .	12
2.2.2. Estándar IEEE 802.15.4 . . . . .	13

2.2.2.1.	Topologías . . . . .	15
	Topología Estrella. . . . .	15
	Topología Peer-to-Peer. . . . .	15
	Topología Mesh. . . . .	16
2.2.2.2.	La Arquitectura . . . . .	16
2.2.2.3.	Modos de Operación de una Red bajo el Estándar IEEE 802.15.4	17
	Modo Beacon Enabled. . . . .	17
	La Supertrama (Superframe). . . . .	18
	Modo Nonbeacon Enabled. . . . .	19
	Modelo de Transferencia de Datos. . . . .	19
2.2.2.4.	Descripción de la Capa PHY . . . . .	19
	Algunos Mecanismos del Estándar IEEE 802.15.4. . . . .	21
	La Unidad de Datos Protocolar de la Capa PHY (PPDU). . .	22
	Unidad de Datos de Servicio de la Capa PHY (PSDU). . .	23
2.2.2.5.	Descripción de la Capa MAC . . . . .	23
	La Unidad de Datos Protocolar de la Capa MAC (MPDU). . .	23
	Tipos de Tramas . . . . .	26
2.2.2.6.	El Proceso de Asociación de un Nodo a la Red (Modo Non- beacon Enable) . . . . .	27
	Asociación Dispositivo - coordinador . . . . .	28
2.2.3.	Las Capas Superiores . . . . .	32
2.2.3.1.	MIWI . . . . .	32
2.2.3.2.	MIWI P2P . . . . .	33
2.2.4.	Protocolo ZigBee . . . . .	33
	Estructura de la NWK . . . . .	33
2.2.4.1.	Formatos de tramas . . . . .	35
	1. Trama de datos . . . . .	35
2.2.4.2.	Tramas de comandos . . . . .	36
	1. Comando de solicitud de rutas . . . . .	36
	2. Comando de aceptación de ruta . . . . .	37
	3. Comando de error de rutas . . . . .	38
	4. Comando de abandono de rutas . . . . .	38
2.2.4.3.	Protocolo de enrutamiento usado en ZigBee . . . . .	39
2.2.5.	Módulos Compatibles con el Estándar . . . . .	40
2.2.5.1.	ZigBit ATZB-24-A2 . . . . .	40
2.2.5.2.	XBee . . . . .	41
2.2.5.3.	XBeePro . . . . .	43
2.2.5.4.	ProFLEX01 . . . . .	44
2.2.5.5.	MRF24J40MB . . . . .	46
2.2.5.6.	MRF24J40MA . . . . .	47
	1. Consideraciones de diseño y montaje . . . . .	48
	2. El patrón de radiación . . . . .	49
	3. Interfaz SPI . . . . .	49
	Lectura y escritura de registros de direccionamiento corto.	49
	Lectura y escritura de registros de direccionamiento largo.	50

4. Configuración . . . . .	51
5. Recepción . . . . .	51
6. Transmisión . . . . .	52
2.2.6. MikroC PRO for PIC V5.6 . . . . .	56
2.2.7. Eagle V6.3 . . . . .	57
2.2.8. El Microcontrolador PIC18F27J53 . . . . .	58
<b>3. El Marco Metodológico</b> . . . . .	<b>62</b>
3.1. Tipo de Investigación . . . . .	62
3.2. Diseño de la Investigación . . . . .	62
3.3. Técnicas e instrumentos de recolección de datos . . . . .	63
3.4. Procedimiento Metodológico . . . . .	63
3.4.1. Fase I - Descripción del estándar IEEE 802.15.4 y selección del protocolo. . . . .	63
3.4.2. Fase II - Diseño del hardware correspondiente a los módulos transceptores. . . . .	64
3.4.3. Fase III - Implementación de un protocolo de transmisión de paquetes. . . . .	65
3.4.4. Fase IV - Desarrollo del algoritmo de la red inteligente. . . . .	66
3.4.5. Fase V - Desarrollar un software de aplicación Windows™ con interfaz gráfica. . . . .	66
3.4.6. Fase VI - Verificación de correcto funcionamiento. . . . .	67
3.4.6.1. Pruebas de Comunicación . . . . .	67
Comunicación: Interfaz - Nodo Principal . . . . .	67
Proceso de Asociación: Nodo Principal - Nodo . . . . .	67
Comunicación: Interfaz - Nodo Principal - Nodo . . . . .	68
Proceso de Asociación: Nodo - Nodo . . . . .	68
Comunicación: Interfaz - Nodo Principal - Nodo Puente - Nodo . . . . .	68
Enrutamiento y listado de errores . . . . .	69
3.4.6.2. Pruebas del Protocolo de la Capa de Aplicación . . . . .	69
<b>4. Los Resultados y su Análisis</b> . . . . .	<b>70</b>
4.1. Descripción del Estándar y la Selección del Protocolo . . . . .	70
4.1.1. Resumen del Estándar IEEE 802.15.4 . . . . .	70
4.1.2. La selección del protocolo de la capa de red . . . . .	71
4.2. Diseño y Fabricación del Hardware . . . . .	71
4.2.1. Diseño de los Dispositivos . . . . .	71
4.2.1.1. Selección del Módulo de Comunicación Inalámbrica . . . . .	71
4.2.1.2. Selección del Microcontrolador . . . . .	73
4.2.1.3. Diseño del Circuito de Rectificación . . . . .	73
4.2.1.4. Diseño del Circuito de Carga de Batería . . . . .	74
4.2.1.5. Consideraciones Varias . . . . .	74
4.2.1.6. Diseño Nodo Principal . . . . .	76
4.2.1.7. Diseño Nodo Sensor ADC Simple . . . . .	79
4.2.1.8. Diseño Nodo Sensor ADC Diferencial . . . . .	81
4.2.1.9. Diseño de Sensor de Temperatura y Presión . . . . .	83
4.2.1.10. Diseño de Tarjeta de Pruebas . . . . .	85

4.2.2.	Fabricación de los Dispositivos . . . . .	85
4.2.2.1.	Costo de los materiales . . . . .	85
	1. Nodo principal: . . . . .	85
	2. Nodo sensor ADC simple: . . . . .	86
	3. Nodo sensor ADC diferencial: . . . . .	86
	4. Sensor de temperatura: . . . . .	87
	5. Sensor de presión: . . . . .	87
	6. Tarjeta de pruebas: . . . . .	88
	7. Costo total del proyecto: . . . . .	88
4.2.2.2.	Dispositivos Terminados . . . . .	88
	1. Nodo principal: . . . . .	88
	2. Nodo sensor ADC simple: . . . . .	89
	3. El nodo sensor ADC diferencial: . . . . .	89
	4. Sensor de temperatura: . . . . .	90
	5. Sensor de presión: . . . . .	90
4.3.	El Protocolo de la Capa de Aplicación . . . . .	90
4.3.1.	Estructura básica . . . . .	91
4.3.2.	Comandos . . . . .	91
4.3.2.1.	Enviar un <i>reset</i> (Comando 0x11) . . . . .	93
4.3.2.2.	Enviar un <i>reset</i> a toda la red (Comando 0x40) . . . . .	93
4.3.2.3.	Respaldar información del coordinador en su memoria flash (Comando 0x44) . . . . .	93
4.3.2.4.	Solicitar información . . . . .	93
	Solicitar ultimo paquete enviado o recibido (Comando 0x12, Co- mando 0x13) . . . . .	93
	Solicitar lista de nodos cercanos (Comando 0x14) . . . . .	94
	Solicitar lista de rutas (Comando 0x15) . . . . .	94
	Solicitar lista de descubrimiento de rutas (Comando 0x16) . . . . .	95
	Solicitar lista de errores (Comando 0x17) . . . . .	95
	Solicitar lista de canales ocupados (Comando 0x18) . . . . .	95
	Solicitar medición ADC (Comando 0x19) . . . . .	96
	Solicitar rango de direcciones para asignación (Comando 0x1A) . . . . .	96
	Lectura de un registro del MRF de corto direccionamiento (Co- mando 0x1B) . . . . .	96
	Lectura de un registro del MRF de largo direccionamiento (Co- mando 0x1C) . . . . .	97
	Solicitar canal de operación a un nodo (Comando 0x1F) . . . . .	97
	Solicitud de la potencia de la antena (Comando 0x21) . . . . .	97
	Solicitar el tipo de sensor (Comando 0x23) . . . . .	98
	Solicitar estado de la batería (Comando 0x24) . . . . .	99
	Solicitar unidad asociada a la medición (Comando 0x25) . . . . .	99
	Solicitar offset o escala de la medición (Comando 0x27, Coman- do 0x29) . . . . .	99
	Solicitar tipo de filtro (Comando 0x2B) . . . . .	100
	Solicitar la cantidad de muestras (Comando 0x2D) . . . . .	100

Solicitar registros de calibración (Comando 0x2F, Comando 0x31)	101
Solicitar tipo de registros de información (Comando 0x33, 0x35, 0x37, 0x39, 0x3B, 0x3D)	101
Solicitar la tasa de muestreo (Comando 0x42)	102
4.3.2.5. Guardado de datos	102
Escritura de un registro del MRF de direccionamiento corto (Comando 0x1D)	102
Escritura de un registro del MRF de direccionamiento corto (Comando 0x1E)	102
Fijar canal de operación de un nodo (Comando 0x20)	102
Fijar potencia de la antena (Comando 0x22)	103
Fijar unidad de medición (Comando 0x26)	103
Fijar offset o escala de medición (Comando 0x28, Comando 0x2A)	103
Fijar la cantidad de mediciones a ser promediadas (Comando 0x2E)	104
Fijar registros de calibración (Comando 0x30, Comando 0x32)	104
Fijar registros de información (Comando 0x33, 0x35, 0x37, 0x39, 0x3B, 0x3D)	104
Fijar canal de operación de la red (Comando 0x41)	104
Fijar intervalo de tiempo entre muestras (Comando 0x43)	105
Fijar fecha y hora (Comando 0x45)	105
4.4. Desarrollo del Algoritmo	105
4.4.1. Algoritmo de Inicio	105
4.4.2. Algoritmo de Interrupciones	105
4.4.3. Algoritmo de Apertura de PAN	107
4.4.4. Algoritmo de Asociación	108
4.4.5. Algoritmo Enviar Paquete	109
4.4.6. Algoritmo Recepción de Solicitud de Ruta	110
4.4.7. Algoritmo Aceptación de Ruta	111
4.4.8. Algoritmo de Medición de Baterías	112
4.4.9. Algoritmo de Listado de Errores	113
Error por falla en la comunicación con un dispositivo cercano	113
Error por falla en la comunicación a través de una ruta	113
Error durante el proceso de enrutamiento	114
4.5. Desarrollo de la Interfaz	114
4.5.1. Depurador de errores	114
4.5.2. Interfaz	116
4.5.2.1. Desarrollo	116
4.6. Pruebas	117
4.6.1. Alcance	117
4.6.2. Consumo de potencia	118
<b>5. Conclusiones</b>	<b>119</b>
<b>Recomendaciones</b>	<b>121</b>

<i>ÍNDICE GENERAL</i>	XV
<b>Bibliografía</b>	<b>123</b>
<b>Glosario</b>	<b>124</b>
<b>Anexos</b>	<b>126</b>
<b>Anexo A</b>	<b>127</b>
<b>Anexo B</b>	<b>130</b>

# Índice de figuras

1.1.	Esquema del Sistema. En verde se muestran las conexiones entre nodos de la red. En rojo se indican dos nodos fuera de rango (A y D). En azul se muestra la ruta para el enlace A-D usando como nodo puente el nodo C. . . . .	4
2.1.	Niveles definidos por la OSI. . . . .	9
2.2.	Topologías que maneja el modelo OSI. . . . .	10
2.3.	Ejemplo de topología tipo estrella. . . . .	15
2.4.	Ejemplo de topología peer-to-peer. . . . .	16
2.5.	Ejemplo de topología mesh. . . . .	17
2.6.	Arquitectura del Estándar IEEE 802.15.4. . . . .	18
2.7.	Estructura de Supertrama. . . . .	19
2.8.	Comunicación dispositivo a coordinador. A la izquierda funcionamiento bajo modo <i>beacon enable</i> a la derecha bajo modo <i>nonbeacon enable</i> . . . . .	20
2.9.	Comunicación coordinador a dispositivo. A la izquierda funcionamiento bajo modo <i>beacon enable</i> a la derecha bajo modo <i>nonbeacon enable</i> . . . . .	20
2.10.	Composición de la capa física (PHY). . . . .	22
2.11.	Estructura básica de la capa MAC. . . . .	23
2.12.	Estructura del campo control de trama ( <i>frame control</i> ). . . . .	24
2.13.	Estructura de un paquete de reconocimiento o <i>acknowledge</i> . . . . .	26
2.14.	Estructura de una trama de guía o <i>beacon</i> . . . . .	27
2.15.	Estructura de un paquete de comando o <i>command</i> . . . . .	27
2.16.	Estructura de un paquete de datos o <i>data</i> . . . . .	28
2.17.	Esquema del proceso de asociación de un nodo a la red funcionando bajo el modo <i>nonbeacon enable</i> . . . . .	29
2.18.	Formato de Comando tipo <i>beacon request</i> . . . . .	30
2.19.	Formato de la petición de asociación. . . . .	31
2.20.	Formato de la confirmación de asociación (Data Request). . . . .	31
2.21.	Formato de la respuesta a la petición de asociación ( <i>association response</i> ). . . . .	32
2.22.	Formato general de la NWK . . . . .	33
2.23.	Control de trama de la capa de red . . . . .	35
2.24.	Trama de datos de la capa de red . . . . .	35
2.25.	Trama de datos de la capa de red . . . . .	36
2.26.	Solicitud de rutas . . . . .	37
2.27.	Aceptación de ruta . . . . .	37
2.28.	Error en ruta . . . . .	38

2.29.	Abandono de ruta . . . . .	39
2.30.	Proceso de solicitud de ruta . . . . .	39
2.31.	Diagrama de de Bloques del módulo ZigBit ATZB-24-A2. . . . .	40
2.32.	Patrón de radiación del módulo ZigBit ATZB-24-A2. A la izquierda se muestra el plano acimutal, en el centro el plano de elevación y a la derecha la referencia con respecto al módulo. . . . .	41
2.33.	Módulo de comunicación inalámbrica XBee. . . . .	41
2.34.	Diagrama de bloques del módulo XBee. . . . .	42
2.35.	Plano acimutal del patrón de radiación de la antena tipo chip del módulo XBee. . . . .	43
2.36.	Módulo de comunicación inalámbrica XBee PRO. . . . .	43
2.37.	Módulo de comunicación inalámbrica ProFlex01. . . . .	44
2.38.	Diagrama de bloques del módulo de comunicaciones ProFLEX01. . . . .	45
2.39.	Módulo de comunicación inalámbrica MRF24J40MB. . . . .	46
2.40.	Planos acimutal (Rojo) y de elevación (Verde) del Patrón de Radiación del módulo MRF24J40MB. . . . .	47
2.41.	Módulo de comunicación inalámbrica MRF24J40MA. . . . .	47
2.42.	Esquema de pines del MRF24J40. . . . .	48
2.43.	Diagrama de diseño y montaje del módulo de comunicaciones <i>MRF24J40MA</i> . . . . .	49
2.44.	Planos acimutal (negro) y de elevación (rojo) del patrón de radiación del módulo de comunicaciones MRF24J40MA. . . . .	50
2.45.	Lectura y escritura de registros de corto direccionamiento del MRF24J40. . . . .	51
2.46.	Lectura y escritura de registros de corto direccionamiento del MRF24J40. . . . .	52
2.47.	Estructura de un paquete recibido por el MRF24J40MA. . . . .	54
2.48.	Secciones de memoria del MRF24J40 para transmisión de los diferentes tipos de trama. . . . .	55
2.49.	Estructura de los diferentes paquetes posibles a ser transmitidos por el MRF24J40MA. . . . .	55
2.50.	Entorno de programación del software MikroC PRO for PIC v5.6. Con el número uno se identifica la barra de herramientas y demás barras de compilación, indentación, entre otros; con el número dos el cuadro de codificación del programa, con el tres la barra de mensajes, con el cuatro el explorador de código y con el cinco las barras de acceso rápido. . . . .	56
2.51.	Entorno de Diseño para Tarjetas de Circuitos Impresos del Software Eagle v6.3. . . . .	58
2.52.	Esquema de pines del PIC18F27J53. . . . .	59
2.53.	Diagrama de Bloques del PIC18F27J53. . . . .	61
4.1.	Circuito de rectificación de la tensión de alimentación. . . . .	74
4.2.	Simulación del circuito de Carga de Batería. . . . .	75
4.3.	Diagrama de bloques del nodo principal de la red. . . . .	76
4.4.	Diagrama de conexiones del PIC18F27J53 para el nodo principal de la red. . . . .	77
4.5.	Diagrama de conexiones del MRF24J40MA para el nodo principal de la red. . . . .	77

4.6.	Diagramas de alimentación y pines de conexión del nodo principal. Se observa en la parte superior el circuito de regulación a 3.3V y el capacitor de 0.1uF del PIC, abajo a la izquierda los pines de programación y a la derecha los pines del puerto USB. . . . .	78
4.7.	Diseño en Eagle de la tarjeta de circuito impreso para el nodo principal. . .	78
4.8.	Diagrama de bloques de los nodos sensores de la red con ADC simple. . .	79
4.9.	Esquema de conexiones del PIC18F27J53 para un nodo sensor. . . . .	80
4.10.	Diagramas del puerto de conexiones del nodo sensor con ADC sencillo. A la izquierda se muestra el esquema de conexiones y a la derecha la vista frontal frontal del puerto de conexiones. . . . .	80
4.11.	Diseño de la tarjeta de circuito impreso para el Nodo Sensor con ADC Simple. . . . .	81
4.12.	Diagrama de Bloques de los nodos sensores de la red con ADC diferencial. . .	82
4.13.	Diagramas del puerto de conexiones del nodo sensor con ADC diferencial. A la izquierda se muestra el esquema de conexiones y a la derecha la vista frontal frontal del puerto de conexiones. . . . .	82
4.14.	Diseño de la tarjeta de circuito impreso para el Nodo Sensor con ADC diferencial. . . . .	83
4.15.	Diagrama de Bloques de los Nodos Sensores de la Red. . . . .	83
4.16.	Esquema circuital de filtros. A la izquierda se muestra el filtro sencillo y a la derecha el filtro para la señal diferencial. . . . .	84
4.17.	Diseño de las tarjetas de circuito impreso para los sensores. A la izquierda se muestra el sensor de temperatura y a la derecha el de presión. . . . .	84
4.18.	Diseño de la tarjeta de circuito impreso para la Tarjeta de Pruebas. . . .	85
4.19.	Nodo principal fabricado. . . . .	89
4.20.	Nodo sensor con ADC simple fabricado. . . . .	89
4.21.	Nodo sensor con ADC diferencial fabricado. . . . .	89
4.22.	Sensor de Temperatura fabricado. . . . .	90
4.23.	Sensor de Presión fabricado. . . . .	90
4.24.	Estructura básica del protocolo . . . . .	91
4.25.	Envío de reset a un dispositivo . . . . .	93
4.26.	Envío de <i>reset</i> a la red . . . . .	93
4.27.	Respaldar información del coordinador . . . . .	93
4.28.	Estructura de solicitud y respuesta del ultimo paquete enviado . . . . .	94
4.29.	Estructura de solicitud y respuesta del ultimo paquete recibido . . . . .	94
4.30.	Estructura de solicitud y respuesta de la lista de nodos cercanos . . . . .	94
4.31.	Estructura de solicitud y respuesta de la lista de rutas . . . . .	95
4.32.	Estructura de solicitud y respuesta de la lista de posibles rutas . . . . .	95
4.33.	Estructura de solicitud y respuesta de la lista de errores . . . . .	95
4.34.	Estructura de solicitud y respuesta de la lista de canales ocupados . . . . .	96
4.35.	Estructura de solicitud y respuesta de la medición . . . . .	96
4.36.	Estructura de solicitud y respuesta del rango de direcciones disponible a asignar . . . . .	96
4.37.	Estructura de solicitud y respuesta de un registro de corto direccionamiento del módulo MRF24j40 . . . . .	97

4.38. Estructura de solicitud y respuesta de un registro de largo direccionamiento del módulo MRF24j40 . . . . .	97
4.39. Estructura de solicitud y respuesta del canal de operación . . . . .	97
4.40. Estructura de solicitud y respuesta de la potencia de transmisión . . . . .	98
4.41. Niveles ajustables de potencia de transmisión del MRF24J40MA. . . . .	98
4.42. Estructura de solicitud y respuesta del tipo de sensor . . . . .	98
4.43. Estructura de solicitud y respuesta del estado de la batería . . . . .	99
4.44. Estructura de solicitud y respuesta de la unidad de medición . . . . .	99
4.45. Estructura de solicitud y respuesta del offset de la medición . . . . .	100
4.46. Estructura de solicitud y respuesta de la escala de la medición . . . . .	100
4.47. Estructura de solicitud y respuesta del filtro digital a usar . . . . .	100
4.48. Estructura de solicitud y respuesta de la cantidad de mediciones que serán promediadas . . . . .	101
4.49. Estructura de solicitud y respuesta de los registros de calibración . . . . .	101
4.50. Estructura de solicitud y respuesta de los registros de información . . . . .	101
4.51. Estructura de solicitud y respuesta del intervalo entre muestras . . . . .	102
4.52. Estructura de la escritura de un registro de direccionamiento corto . . . . .	102
4.53. Estructura de la escritura de un registro de direccionamiento largo . . . . .	102
4.54. Estructura de la escritura del canal . . . . .	103
4.55. Estructura del cambio de la potencia de transmisión . . . . .	103
4.56. Estructura del cambio de la unidad de medición . . . . .	103
4.57. Estructura del cambio del offset . . . . .	103
4.58. Estructura del cambio de la escala . . . . .	103
4.59. Estructura del cambio del filtro . . . . .	103
4.60. Estructura del cambio de la cantidad de mediciones a ser promediadas . . . . .	104
4.61. Estructura del cambio de los registros de calibración . . . . .	104
4.62. Estructura del cambio de los registros de información . . . . .	104
4.63. Estructura del cambio del canal de la red . . . . .	104
4.64. Estructura del cambio de la cantidad de mediciones a ser promediadas . . . . .	105
4.65. Estructura del cambio de la fecha . . . . .	105
4.66. Diagrama de bloques del algoritmo de inicio del código del microcontrolador.	106
4.67. Diagrama de bloques del algoritmo de interrupciones. . . . .	107
4.68. Diagrama de bloques del algoritmo de apertura de red. . . . .	108
4.69. Diagrama de bloques del algoritmo de asociación a la red. Se marcan de color naranja los diferentes puntos de acceso al algoritmo, dependiendo del paquete perteneciente al proceso de asociación que se reciba. . . . .	109
4.70. Proceso de solicitud de rutas . . . . .	110
4.71. Proceso de recepción de solicitud de ruta . . . . .	111
4.72. Diagrama de bloques del algoritmo de Aceptación de Ruta. . . . .	112
4.73. Diagrama de bloques del algoritmo de medición de batería. . . . .	113
4.74. Diagrama de bloques del algoritmo de listado de errores. . . . .	114
4.75. Depurador de errores . . . . .	115
4.76. Vista de mediciones sin listas de errores . . . . .	116
4.77. Vista de mediciones con listas de errores . . . . .	117
4.78. Vista de Información adicional . . . . .	117

5.1. Diagrama funcional del coordinador de la red . . . . .	127
5.2. Diagrama funcional de los nodos . . . . .	128
5.3. Puerto de conexiones del nodo con conversor analógico-digital simple . . . .	129
5.4. Puerto de conexiones del nodo con conversor analógico-digital diferencial . .	129

# Índice de cuadros

2.1.	Algunos parámetros concernientes a la capa física (PHY) perteneciente al estándar 802.15.4. . . . .	21
2.2.	Valores del subcampo tipo de trama ( <i>frame type</i> ). . . . .	24
2.3.	Valores posibles para los subcampos <i>destination addressing mode</i> y <i>source addressing mode</i> del campo <i>frame control</i> . . . . .	25
2.4.	Comandos asociados a la trama <i>command</i> . . . . .	28
2.5.	Estados de asociación posibles para el campo <i>status</i> . . . . .	32
2.6.	Dispositivos Zigbee . . . . .	34
2.7.	Valores del campo de tipo de trama . . . . .	34
2.8.	Tipos de comandos de la capa de red . . . . .	36
2.9.	Campo de opciones de comandos . . . . .	37
2.10.	Códigos de error . . . . .	38
2.11.	Configuración de los parámetros de control del MRF24J40MA para un dispositivo Coordinador de PAN. . . . .	53
3.1.	Actividades para la descripción de las características del estándar IEEE 802.15.4 y la selección del protocolo para las capas superiores. . . . .	64
3.2.	Descripción de las actividades realizadas para el diseño y fabricación de los dispositivos. . . . .	65
3.3.	Descripción de las actividades realizadas para el desarrollo del protocolo de la capa de aplicación. . . . .	65
3.4.	Descripción de las actividades realizadas para el desarrollo del algoritmo del microcontrolador. . . . .	66
3.5.	Descripción de las actividades realizadas para el desarrollo de la interfaz gráfica de usuario. . . . .	66
4.1.	Comparación entre los protocolos MIWI, MIWI P2P y ZigBee. . . . .	71
4.2.	Cuadro comparativa de los módulos de comunicación inalámbrica compatibles con el estándar IEEE 802.15.4 bajo estudio. . . . .	72
4.3.	Costos de componentes para la fabricación del Nodo Principal. . . . .	85
4.4.	Costos de componentes para la fabricación del Nodo Sensor ADC Simple. . . . .	86
4.5.	Costos de componentes para la fabricación del Nodo Sensor ADC Diferencial. . . . .	87
4.6.	Costos de componentes para la fabricación del sensor de temperatura. . . . .	87
4.7.	Costos de componentes para la fabricación del sensor de presión. . . . .	88
4.8.	Costos de componentes para la fabricación de la tarjeta de pruebas. . . . .	88

4.9. Costo total del proyecto. . . . .	88
4.10. Cuadro de comandos del protocolo de comunicación para la capa de aplicación. . . . .	91
4.11. Continuación de la tabla de comandos del protocolo de comunicación para la capa de aplicación. . . . .	92
4.12. Unidades de medición . . . . .	99

# Introducción

Hace unas cuantas décadas el desarrollo de la tecnología mostraba, en su crecimiento, un ritmo relativamente lento. Los avances tecnológicos eran contemplados además, por un grupo selecto de personas y la gran mayoría de la población permanecía ignorante de estos nuevos y grandes logros de la ciencia, sin que esto afectara su vida cotidiana. Con el pasar del tiempo, la tecnología ha tomado campo en nuestro día a día, involucrándose en nuestra vida cotidiana, logrando solventar problemas básicos de nuestra existencia, lo que nos permite realizar nuestras actividades de una forma más cómoda, más sencilla y más simple. Estas comodidades, a las que nos hemos acostumbrando rápidamente, le han proporcionado grandes cambios a nuestra existencia y nos han llevado hasta un punto en el que no nos imaginamos ya la vida sin ellas. Con el pasar de los años, algunos de estos avances tecnológicos como el televisor, las computadoras personales y el teléfono celular, entre muchos otros, han convertido a nuestra sociedad en una sociedad demandante y exigente; en una sociedad que anda siempre en busca de lo nuevo, de lo último; en una sociedad que vive ahora ávida de conocer los nuevos avances de la tecnología y que está siempre atenta al desarrollo de las cosas al filo de la vanguardia.

Este nuevo y acelerado ritmo de crecimiento de la tecnología forzó al mercado a evolucionar de una forma exponencial para poder cubrir la demanda de este nuevo e insaciable consumidor. Así, hoy en día, sin darnos cuenta, vivimos en un mundo tecnológicamente cambiante, rodeados de avances tecnológicos en todos los niveles y campos relativos a nuestra existencia, pero que se ha involucrado con la misma velocidad arrolladora en todos los aspectos de la humanidad, ayudando a la ciencia a obtener respuestas que antes eran consideradas como imposibles.

El mundo de las comunicaciones no es la excepción a la regla. En la actualidad nos hemos acostumbrado a estar siempre en contacto con nuestro entorno, satisfaciendo también nuestra necesidad de información y conocimiento, sobre todo a través de ese gran salto tecnológico que ha representado la Internet para el desarrollo de la humanidad. Las redes sociales son un claro ejemplo de esta nueva situación, ya que casi todo en nuestra vida cotidiana está de una u otra forma vinculado con redes de comunicación.

En este aspecto de la tecnología vemos además como la industria y el comercio no se han quedado atrás. han ido integrando la tecnología en todos sus procesos de control y procedimientos. Desde los más básicos y sencillos como los controles de acceso y las alarmas, hasta los muy elaborados procesos de ensamblaje automatizados. Sin importar cual sea la especialidad a la que pertenezca, toda la industria requiere de procesos para el manejo de múltiples variables, logrando este objetivo mediante el uso de sistemas de medición conformados por redes de sensores, conectados estos a centrales de medición a través de cables de instrumen-

tación. Estos cableados requieren a su vez de instalaciones numeradas, identificadas, además de tuberías, canaletas y una gran cantidad de materiales asociados a esta integración. Todo ello representa un costo que se vuelve cada vez más oneroso y su implementación se convierte en una complicación para todo aquel que persiga o requiera una mejora en este aspecto de la automatización de los procesos, las mediciones y los controles. Entonces, surgieron las inquietudes: ¿Por qué no pensar en la integración de estos sistemas a una red inalámbrica? ¿Por qué no cambiar todo este engorroso y costoso sistema de cableado y seguir evolucionando hacia una industria más eficiente a través de este tipo de comunicación?.

Por lo antes expuesto, el objetivo de este trabajo especial de grado fue el de desarrollar un prototipo de red inalámbrica conformada por nodos que, sin importar la naturaleza de la medición, fuesen capaces de transmitirla a través de dicha red y hacerlo de manera inteligente. Desarrollar una red que para el usuario fuese de fácil instalación y operación, con un alto nivel de confiabilidad y, además, con un costo considerablemente menor al de las otras opciones disponibles.

# Capítulo 1

## El Problema

### 1.1. Planteamiento del Problema

Los seres humanos siempre han tenido el deseo de extender sus capacidades, como lo son las habilidades mecánicas, con la revolución industrial, las habilidades intelectuales, con el desarrollo computacional, y con la invención de sensores, los humanos expanden, nuevamente, sus habilidades que, junto a los desarrollos antes mencionados, permiten sensor, procesar datos y tomar acciones para el control de un proceso cualquiera.

Creed Huddleston comenta que vivimos en un mundo de "acceder instantáneamente" a todo, donde la gente desea la información que quiere, en la forma que quiere y a un precio aceptable (preferiblemente gratis); con una economía donde, sin importar el producto, lo más escaso es lo más valorado (y más costoso). Se introduce entonces, el hecho de que los sensores no son la excepción [1].

Este deseo de expandir las capacidades humanas para percibir el mundo ha generado un incremento veloz en el desarrollo y venta de los sensores, tal como se expresa en las investigaciones de mercado del reporte IAS006D "Sensors: Technologies and Global Markets"[2] de "BCC Research", tan solo en el 2010 el mercado de los sensores generó un movimiento monetario de aproximadamente 56.3 mil millones de dolares, con un crecimiento anual esperado del 7.8 %, dando una estimación de crecimiento 35.2 mil millones de dolares, aproximadamente, para el 2016. De todo este mercado, la participación de Venezuela, es casi nula.

Los sensores convencionales (cableados y sin inteligencia), usualmente necesitan estar físicamente cerca del sistema de control y monitoreo que recibe sus mediciones. En general, mientras el sensor esté a mayor distancia, este necesitará de una mayor cantidad de cable y por ende, representa un mayor costo de instalación, en especial cuando requiere de un cable especializado (instrumentación) para su conexión. Aún peor es el hecho de que la calidad de la medición se degenera a medida que aumenta dicha distancia [1].

Sin embargo, existen principalmente dos opciones a la hora de enviar la información medida: la convencional, que implica el cableado del sistema de medición, donde la central de monitoreo realiza el procesamiento de dichas mediciones, o se puede añadir un microcontrolador que le dé al sensor la inteligencia y la capacidad de procesar dichas mediciones. Además al agregar, de alguna manera, un enlace inalámbrico entre el sensor y el sistema de monitoreo, se solventa el problema de la degeneración de la medición. Claro está que un microcontrolador no posee la capacidad para manejar la gran cantidad de información que

una central de monitoreo puede manejar; pero ofrece un sistema flexible y de menor costo [1]. Por esta razón, en la última década se ha experimentado un auge en el desarrollo de sensores inteligentes<sup>1</sup>.

Ahora, dicha inteligencia puede llevarse a un nivel más alto y permitir a dos sensores comunicarse entre ellos (no necesariamente con la central), formar una red, llevar información a través de rutas, entrar de forma automática en modos de bajo consumo, entre otros.

Es por esto que se desea realizar, como trabajo especial de grado, el diseño de una red inalámbrica, basada en el estándar IEEE 802.15.4, de sensores inteligentes, dinámica, y que sea capaz de detectar automáticamente nodos en las cercanías, crear rutas, recuperarse de fallos de forma espontánea; una red que implemente la tecnología de microcontroladores PIC como recursos de procesamiento, que posea una alta fiabilidad.

## 1.2. Justificación de la Investigación

Actualmente, Venezuela tiene una participación poco importante dentro de la producción de sensores a nivel mundial, por lo que el presente trabajo de grado, representa un aporte tecnológico para el desarrollo de la nación, permitiéndole incurrir en un mercado sumamente lucrativo, como lo es el de los sensores.

La industria se caracteriza por contener dentro de sus procesos de producción, un gran número de dispositivos de medición o sensores que le permite ejercer el control de dicho proceso. Sin embargo, al implementar la red desarrollada, se reducen los costos del sistema de medición, ya que no se hace necesaria la planificación y construcción de sistemas de canalización, como la instalación tanto de cableado como de los dispositivos. Además, esta red cuenta con una inteligencia que le permite establecerse automáticamente sin que el operador deba realizar ningún procedimiento adicional, acortando el tiempo de puesta en marcha de cualquier nueva sucursal de cualquier empresa en vías de expansión.

Además representa un aporte significativo para la Universidad de Carabobo y su estudiantado, ya que abre una rama de estudio en cuanto al desarrollo de redes inalámbricas dentro de la Facultad de Ingeniería.

Usualmente los sensores que se consiguen en el mercado, son fabricados con una finalidad en particular, es decir, cada sensor está diseñado para medir un tipo de variable específica (presión, temperatura, etc.). Los nodos que constituyen la red propuesta, fueron desarrollados para actuar como interfaz inalámbrica, sin importar la naturaleza de la medición, debido a que cuentan con un número de entradas digitales y analógicas que les permite recibir cualquier dato o tomar cualquier medición que sea necesaria.

---

<sup>1</sup>El término sensor inteligente proviene del inglés *smart sensor*, el cual hace referencia a un dispositivo de medición con ciertas aptitudes como: procesar la información de medición (cálculos, unidades, entre otros), contar con algoritmos especializados que le permita realizar tareas un tanto más complejas, entre otras, siempre con un relativo nivel de sencillez. Este término no debe confundirse con algoritmos de inteligencia artificial ni ningún otro similar.

## 1.3. Los Objetivos de la Investigación

### 1.3.1. Objetivo General

- Desarrollar un prototipo de red de sensores inteligentes inalámbricos basado en el estándar IEEE 802.15.4 para uso en ambientes industriales.

### 1.3.2. Objetivos Específicos

- Describir las capas físicas y de control de acceso al medio del estándar IEEE 802.15.4 para comprender el funcionamiento de los dispositivos compatibles con dicho estándar.
- Seleccionar el protocolo de comunicación para la capa de red dentro de los disponibles como lo son MiWi, MiWi P2P y ZigBee.
- Diseñar el hardware correspondiente a los módulos transreceptores seleccionando los componentes de menor consumo de potencia con la finalidad de brindarle a dichos módulos la resistencia necesaria para soportar el entorno industrial y optimizar su rendimiento energético.
- Implementar un protocolo de transmisión de paquetes de datos que permita el enlace entre los nodos de la red inalámbrica además de la incorporación de un código de verificación por redundancia cíclica para comprobar la correcta transmisión de los datos de cada comunicación.
- Desarrollar un algoritmo que le brinde a cada nodo la capacidad de detección, comunicación y creación de rutas entre los nodos de la red.
- Crear una interfaz de usuario que permita la configuración de los parámetros de operación de los sensores, además de la medición en tiempo real de parámetros relativos al estado del sensor inalámbrico.

## 1.4. Alcance y Delimitación

En el presente trabajo se contempla el diseño de cuatro dispositivos, tres funcionando como sensores y el otro funcionará como punto de enlace entre el computador y la red (Nodo Principal). Dicha red estará fundamentada en el uso de dispositivos compatibles con el estándar IEEE 802.15.4.

En la Figura (1.1) se muestra un esquema del sistema desarrollado, donde se puede observar, de manera generalizada las características funcionales básicas de este.

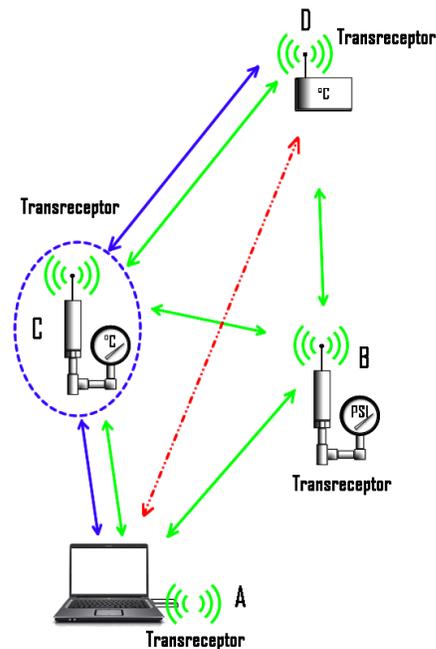


Figura 1.1: Esquema del Sistema. En verde se muestran las conexiones entre nodos de la red. En rojo se indican dos nodos fuera de rango (A y D). En azul se muestra la ruta para el enlace A-D usando como nodo puente el nodo C.

El nodo principal consta de:

- Módulo de comunicación RF compatible con el estándar IEEE 802.15.4
- Microcontrolador de la gamma 18 de bajo consumo de potencia
- Puerto USB para comunicación y alimentación.
- Circuito de regulación de tensión.

Los sensores constarán de:

- Módulo de comunicación RF compatible con el estándar IEEE 802.15.4
- Microcontrolador de la gamma 18 de bajo consumo de potencia.
- Memoria para almacenamiento de mediciones, rutas de conexión entre nodos, parámetros de medición, entre otros.
- Sensor de temperatura o de presión según sea el caso.
- Alimentación dual entre energización alámbrica con tensión DC y baterías.
- Circuito de recarga para baterías.

Cada uno de los módulos o nodos de la red tienen la capacidad de detectar a otros que estén dentro de su alcance, con la finalidad de establecer la red de forma automática, sin necesidad de ningún tipo de configuración previa, además de registrarlos en memoria en caso de fallos en la red poder recuperarse y volver a su estado previo. La red es capaz de enrutar paquetes mediante procesos "multi-salto" de hasta dos nodos intermedios (no es posible establecer comunicación entre nodos que requieren más de dos nodos puente).

La red es capaz de buscar rutas alternas en caso de salida de operación de algún nodo que forme parte de alguna ruta. Esto permite que la red permanezca activa frente a fallos. Además, registra los errores ocurridos en memoria con fecha y hora de detección.

Entre otras cosas, la red es capaz de recuperar nodos que de alguna forma han perdido el contacto con el nodo principal impidiendo que algún nodo se quede aislado de la red.

La programación de los microcontroladores se realizó mediante el compilador de lenguaje C denominado MikroC for PIC v5.6 desarrollado por Mikroelectrónica.

Se utilizará el software Visual Studio 2010 para crear la interfaz de usuario que permita tener acceso a la red y visualizar las rutas de acceso que hayan establecido los nodos, niveles de carga en baterías y cargar los reporte de fallo registrados por cada uno de ellos.

# Capítulo 2

## El Marco Teórico

### 2.1. Antecedentes

Dentro de los trabajos elaborados que siguen la misma línea de investigación planteada se encuentran los siguientes:

- ZigBee Alliance (2004), "ZigBee Specification" [3], manual de especificaciones de las capas de Red y de Aplicación, publicado por "ZigBee Alliance". Tiene como objetivo ofrecer una clara guía de instrucciones a la hora del desarrollo de software basado en el protocolo ZigBee y/o el entendimiento de los procesos internos de dispositivos que funcionen bajo este protocolo. En esta publicación se muestran en detalle los procesos de enrutamiento, formación de rutas con la mejor calidad de enlace, detección de nodos cercanos, procesos de respaldo en memoria de las listas de dichas rutas, bajo qué condiciones debe eliminarse un nodo de dicha tabla, la herramienta Radius, entre otros. Se tomó de esta publicación como guía en los procesos de enrutamiento y creación de listas de rutas así como detección de nodos cercanos y su alistamiento en memoria para el desarrollo del código del microcontrolador.
- Tim Cutler (2005), "Implementing ZigBee Wireless Mesh Networking" [4], artículo que describe la implementación de la topología Mesh con las capas de red y aplicación de ZigBee. Tiene como objetivo la descripción de los parámetros que comprenden el manejo de esta topología, cuantificar los niveles de potencia que se manejan típicamente por los dispositivos dentro de una red inalámbrica de área personal de baja transferencia (LR-WPAN) y describir las principales aptitudes que deben tener dichos dispositivos según su jerarquía dentro de la red. Se tomó este artículo como referencia para el diseño de la estructura de la red, las capacidades y aptitudes que debe tener cada nodo según su jerarquía dentro de ella, así como los ajustes de potencia de cada uno de los dispositivos según la antes mencionada jerarquía.
- Eady Fred (2007), "Hands-On ZigBee Implementing 802.15.4 with Microcontrollers" [5], libro publicado por la editorial Elsevier, donde se describen todos los aspectos necesarios para el manejo de la capa de red. Dentro de su desarrollo el autor realiza un proceso experimental para mostrar las capas MAC (inferior) y la de red mediante el uso de dos módulos que se comunican entre sí y un tercer módulo denominado "Sniffer", el

cual recibe los paquetes en el aire de ambos dispositivos, permitiendo la observación, al detalle, de los procesos de asociación de un nodo a la red. Lo recibido por el Sniffer es luego descrito bit a bit por el autor. El objetivo de este proceso es describir el manejo del octeto "Control de Trama" (frame control) de la Capa MAC y su configuración para cada tipo de trama enviada en el proceso de asociación, además de la estructura de cada tipo de paquete transmitido durante este. El autor concluye exitosamente el proceso de asociación y deja definido los pasos a seguir para el ensamblaje de cada paquete según la trama bajo la cual será transmitido cada uno. De igual forma delimita el diagrama de flujo del proceso de asociación para su mejor comprensión. Para este trabajo de grado se tomó la descripción de la estructura de cada trama en el proceso de asociación (*beacon request*, *beacon*, entre otros que serán descritos más adelante) para el desarrollo del código del microcontrolador, el cual se encarga de procesar la información recibida por el módulo de comunicación inalámbrica, procesarla y cargar nuevamente la respuesta al paquete recibido en dicho módulo RF para ser transmitida.

- Dogan Ibrahim (2008), "Advanced PIC Microcontroller Projects in C - From USB to ZigBee with the PIC 18F Series" [6], libro publicado por la editorial Elsevier, donde se describen procesos de medición con sensores de temperatura y presión. De igual forma se describe la utilización de bibliotecas en lenguaje C para el uso del módulo USB del microcontrolador, además de la implementación de esta gama como control para el protocolo ZigBee, presentando bases para la implementación de la serie 18F como microcontrolador host para los dispositivos de una red de sensores inalámbricos. Se tomó esta publicación como base para sentar la factibilidad de la implementación del PIC18F27J53 como microcontrolador host de los protocolos de comunicación y verificación de su posibilidad para el soporte de la red planteada para este trabajo de grado.
- Caverzán Franco, Vargas Carlos (2011), trabajo especial de grado de ingeniería eléctrica, Universidad de Carabobo "Desarrollo de un sistema de monitoreo inalámbrico del nivel en tanques de almacenamiento de líquidos basados en dispositivos compatibles con el estándar IEEE 802.15.4" [7]. Como su título lo indica el objetivo de este trabajo de grado fue el de desarrollar y fabricar un sistema de monitoreo inalámbrico que les permitiera comunicar un dispositivo capaz de medir los niveles de fluido en una serie de tanques, con uno central mediante el uso de un par de módulos de comunicación inalámbrica compatibles con el estándar IEEE 802.15.4. Para ello fue necesario el estudio de una serie de módulos de comunicación compatibles con dicho estándar, estudio que descarta al MRF24J40MA (Microchip) por este último no contar con un microcontrolador que hiciera el manejo automático del estándar en cuestión (otros módulos, de mayor costo, contaban con un microcontrolador que automáticamente realizaban los procesos de asociación y demás procesos concernientes al estándar) y donde no fueron considerados criterio de costo ni consumo de potencia. Para este trabajo de grado se tomó el estudio comparativo de los módulos RF como base y se extendieron los criterios de selección con los anteriormente mencionados, implicando esto la profundización de la información pertinente a cada uno de los módulos estudiados para su posterior comparación y finalmente la selección del módulo de comunicación inalámbrica a implementar en este trabajo de grado. De igual forma se hizo necesario, por parte de Caverzán y

Vargas, el diseño de un protocolo de comunicación para la capa de aplicación que les permitiera la transferencia de información entre ambos dispositivos; protocolo que demostró su efectividad al permitir dicha comunicación de manera práctica y de la forma más sencilla posible. Se tomó para el presente trabajo de grado dicho protocolo como guía para el desarrollo de uno propio, con un número mayor de comandos que permita la fácil configuración de la red y que ofrezca básicamente las mismas características del desarrollado en el trabajo de grado en cuestión.

## 2.2. Bases Teóricas

### 2.2.1. Modelo OSI

Establecida en 1947, la Organización Internacional de Estándares, mayormente conocida como ISO por sus siglas en inglés, introdujo en 1970 el modelo de Interconexión de Sistemas Abiertos u OSI (Open System Interconnection) bajo el nombre de ISO/IEC 7498-1, el cual fue propuesto como capaz de cubrir todos los aspectos concernientes a comunicaciones en red.

Primeramente es conveniente definir el concepto de "sistema abierto" para el mejor entendimiento de la descripción de la pila que conforma este modelo, como un conjunto de protocolos que hacen posible la comunicación entre dos sistemas sin importar la arquitectura subyacente de cada uno.

A su vez es de interés el resaltar que el propósito de este modelo es el de mostrar como facilitar la comunicación de los sistemas antes mencionados sin que ninguno de ellos deba incurrir en cambios en hardware y/o software, por lo que fue desarrollado para el entendimiento y diseño de una arquitectura de red que sea flexible, robusta e inter-operable, proponiendo un conjunto de siete(7) capas como se muestra en la Figura 2.1, donde cada una define una parte del proceso de transporte de información a través de la red [8].

Dichas capas son descritas a continuación:

#### 2.2.1.1. Capa Física

La capa física coordina las funciones requeridas para el transporte de un conjuntos de bits a través del medio físico. De igual forma, define todos los procedimientos y funciones que los dispositivos físicos deben llevar a cabo para que una transmisión pueda ocurrir [8]. Además la capa física regula puntos como:

- a) **Características Físicas de la Interfaz y el Medio.** La capa física define las características de la interfaz entre los dispositivos y el medio de transmisión como frecuencia de operación, ancho de banda, tipo de conectores (redes alámbricas), tipo de modulación (redes inalámbricas), entre otros. De igual forma define el tipo de medio de transmisión como por ejemplo: guías de onda (cable coaxial, bifilar, par trenzado, fibra óptica, etc.), aire (redes inalámbricas), entre otros.
- b) **La representación de los bits.** La información o data que maneja la capa física consiste en cadenas de bits que deben ser codificados, ya sea en señales ópticas o eléctricas, para poder ser transmitidos. De igual forma define el tipo de codificación



Figura 2.1: Niveles definidos por la OSI.  
Referencia: [8]

a implementar para dicho proceso de codificación, es decir, la forma en como los bits serán convertidos en señales.

- c) **El ancho de banda de transmisión.** El número de bits por segundo que se transmiten está regulado por la capa física. Además de los niveles en amplitud de la señal (que nivel define un "uno" y que otro un "cero" además de la duración de cada uno) y a su vez, regula todos los aspectos de la sincronización, es decir, verifica que tanto el emisor como el receptor además de tener el mismo nivel de transferencia (bits por segundo), posean sus relojes en sincronismo (los procesos de comunicación se realizan de forma sincrónica).
- d) **Configuración de canal.** La capa física está encargada de conectar cada dispositivo con el medio. En una configuración "punto a punto" la capa física comunica a ambos mediante un canal, mientras que en una "multipunto" todos los puntos compartirían el mismo canal.
- e) **Topología.** La capa física define como los dispositivos se conectan para formar una red. Estas pueden ser de tipo "Mesh" (cada dispositivo puede conectarse con cualquier otro en la red), tipo estrella (todos los dispositivos se conectan a un dispositivo central), tipo anillo (cada dispositivo se conecta con el siguiente hasta formar un anillo), tipo "bus" (todos los dispositivos se enlazan bajo el mismo enlace) o híbrida (combinación de una o más topologías de red). En la Figura 2.2 se muestran ejemplos gráficos de cada topología.
- f) **Modo de transmisión.** La capa física de igual forma define la dirección de comunicación entre dos dispositivos: simplex (sólo un dispositivo es capaz de transmitir y el otro sólo de recibir), half-duplex (ambos pueden transmitir y recibir pero no al mismo tiempo) o full-duplex (Ambos pueden transmitir y recibir al mismo tiempo).

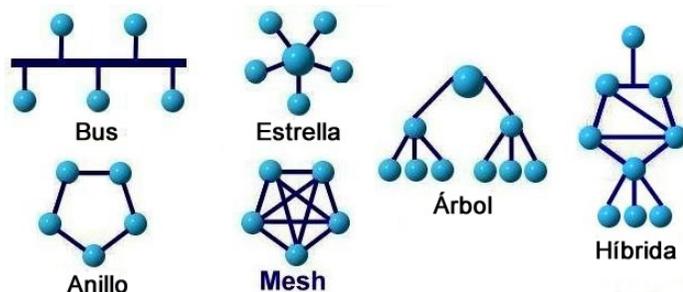


Figura 2.2: Topologías que maneja el modelo OSI.

### 2.2.1.2. Capa de Enlace de Datos

La capa de enlace de datos es la encargada de convertir a la capa física, la cual funciona como un emisor-receptor de fuerza bruta carente de interpretación alguna, en un enlace confiable de alto nivel [8]. Además es responsable de un número de procesos definidos a continuación:

- a) **Entramado.** La capa de enlace de datos divide las cadenas de bits recibidas de la capa de red, en unidades de información manejables para la capa física, llamadas tramas.
- b) **Direccionamiento Físico.** Si diferentes tramas necesitan ser enviadas a distintos sistemas dentro de la red, la capa de enlace de datos es la que está encargada de añadir un encabezado a cada trama que defina qué nodo es el remitente y cuál otro el destinatario (cada uno en el caso de ser necesario) para el posterior entendimiento y procesamiento de los paquetes por parte de los diferentes receptores.
- c) **Regulación del Ancho de Banda.** Si la velocidad con la que se están recibiendo los datos por cualquier circunstancia, llegara a ser menor que la velocidad con la que el transmisor la está enviando, la capa de enlace de datos se encarga de limitar esta velocidad de transferencia para evitar saturar al receptor.
- d) **Control de Errores.** La capa de enlace le añade confiabilidad a la capa física, mediante la adición de mecanismos de detección y retransmisión de paquetes dañados o perdidos. De igual forma son detectados paquetes duplicados.
- e) **Control de Acceso.** Cuando dos o más dispositivos comparten un enlace, los protocolos de la capa de Enlace de Datos son los que regulan cuando cada uno de los dispositivos puede o no hacer uso del canal en cierto instante de tiempo.

### 2.2.1.3. Capa de Red

La capa de red, es la responsable de entregar un paquete del emisor al receptor, incluyendo la posibilidad de tener, este último, que atravesar múltiples redes. Cabe destacar que si dos dispositivos se encuentran conectados al mismo canal y son parte de la misma red, no es necesaria la utilización de esta capa, sin embargo, esta capa es imprescindible cuando el emisor y el receptor forman parte de redes diferentes.

También tiene como parte de sus responsabilidades los siguientes aspectos:

- a) **Direccionamiento Lógico.** El direccionamiento realizado en la capa de enlace de datos, solvente el problema de resolución de identificadores de red a nivel local pero al ser necesario el envío de un paquete a otra red será imprescindible un sistema adicional que permita la diferenciación entre el sistema emisor y el receptor. Por ello la capa de red añade a cada paquete proveniente de las capas superiores, un encabezado que incluya las direcciones de la red emisora y la red receptora.
- b) **Enrutamiento.** La capa de red provee un mecanismo de enrutamiento que le permite al paquete ser transportado hasta su destino final.

#### 2.2.1.4. Capa de Transporte

La capa de red percibe cada paquete como un mensaje individual que tiene un remitente y un destinatario, esta no percibe relación alguna entre ninguno de ellos, sin embargo, la realidad es que cada paquete puede o no formar parte de un bloque de información aún más grande. La capa de transporte es la encargada de verificar de que todos los paquetes lleguen completos y en orden al destinatario. Esto lo logra haciendo uso de los mecanismos de control de errores y la regulación de ancho de banda mencionados en la descripción de la capa de red [8].

Otras responsabilidades de la capa de transporte son:

- a) **Direccionamiento de Service-Point.** Las computadoras usualmente mantienen varios procesos activos a la vez. Por esto, no sólo será necesario el envío de paquetes de computadora a computadora, sino también es el especificar a que proceso específicamente se dirige el paquete. Por lo que la capa de transporte, le añade al paquete un direccionamiento llamado Service-Point. De esta manera, la capa de red hace llegar el paquete al computador de destino y la capa de transporte al proceso de destino.
- b) **Segmentación y Reensamblaje.** Como se mencionó anteriormente, la información (de ser suficientemente extensa) debe ser dividida en paquetes que permitan su transmisión. La capa de transporte es la encargada de este proceso. Al hacerlo, le asigna, a cada paquete, un identificador llamado el "Número de Secuencia", número que incrementa de uno en uno con cada transmisión. De esta forma al ser recibidos estos pueden ser ordenados y reensamblados para formar el bloque dividido inicialmente. Cabe destacar que el proceso de reensamblaje le compete de igual forma a la capa de transporte.
- c) **Control de Ancho de Banda.** Al igual que la capa de enlace de datos, la capa de transporte posee una regulación de ancho de banda, que a diferencia de la de la capa de enlace de datos (punto a punto), esta regula el ancho de banda o velocidad de transferencia entre el emisor y el destino final.
- d) **Control de Errores.** De igual forma que la pauta anterior la capa de transporte tiene un mecanismo de detección de errores pero a diferencia de la capa de enlace de datos, en la capa de transporte los errores son verificados de proceso a proceso.

### 2.2.1.5. Capa de Sesión

Para algunos procesos no son suficientes los servicios prestados por las capas anteriormente descritas por lo que la capa de sesión es la controladora del diálogo en la red. Es de su competencia lo relacionado al establecimiento, mantenimiento y sincronismo de la interacción en sistemas de comunicación [8]. Las responsabilidades específicas de la capa de sesión son:

- a) **Control de Diálogo.** La capa de sesión permite que dos sistemas puedan entrar en diálogo (lograr la comunicación entre dos procesos tanto en modo Half-Duplex como Full-Duplex).
- b) **Sincronización.** La capa de sesión posee un proceso que permite añadir puntos de control cada cierta cantidad de paquetes para llevar un control de aquellos que ya han sido transmitidos de tal forma de que si, por alguna circunstancia, es interrumpida la comunicación, no sea necesario retransmitir todos los paquetes nuevamente.

### 2.2.1.6. Capa de Presentación

La capa de presentación es la que se encarga del manejo de la sintaxis y semántica de la información intercambiada entre dos sistemas [8]. Las responsabilidades específicas de la capa de presentación son:

- a) **Traducción.** Este servicio se encarga de convertir la información del formato en el cual esta originalmente es manejada por el dispositivo emisor a un formato común, entendible para cualquier sistema. De igual forma, es responsabilidad de esta capa, del lado receptor, volver a convertir la información a un formato que este pueda manejar.
- b) **Cifrado y Descifrado.** Este servicio permite mantener la privacidad de la comunicación ya que codifica y decodifica la información al ser transmitida y recibida respectivamente.
- c) **Compresión.** Este servicio permite reducir el número de bits a transmitir, hecho particularmente importante en la transmisión de archivos multimedia como lo son texto, audio y video.

### 2.2.1.7. Capa de Aplicación

Permite al usuario acceder a la red. Provee de interfaces y soporte para servicios tales como correo electrónico, acceso remoto a archivos, manejo de bases de datos, entre otros [8]. Los servicios que provee la capa de aplicación incluyen los siguientes:

- a) **Terminal Virtual de Red.** Este servicio le permite al usuario crear una copia virtual de un puerto físico en un host remoto, esto le permite registrarse en dicho host.
- b) **Transferencia, acceso y manejo de archivos.** Este servicio le permite al usuario, mover, enviar, editar archivos en un host remoto, entre otros.
- c) **Servicios de e-mail.** Este servicio proporciona las bases para el manejo de correos electrónicos y su almacenamiento.

- d) **Servicios de directorio.** Proporciona las fuentes para bases de datos de distribución y el acceso a información global concernientes a diversos objetos y servicios.

### 2.2.2. Estándar IEEE 802.15.4

En el año 2003, el grupo de trabajo IEEE P802.15, el cual forma parte del proyecto 802 de la IEEE (Institute of Electrical and Electronics Engineers), introdujo un estándar, para regular los parámetros concernientes a la capa física (PHY) y la capa de control de acceso al medio (MAC) dentro de una red inalámbrica de área personal de baja transferencia o LR-WPAN, por sus siglas en inglés (Low Rate - Personal Area Network).

Primeramente, y antes de entrar de lleno con la descripción del estándar, es necesario delimitar previamente algunos tópicos.

Una LR-WPAN, puede describirse como una red de bajo costo que permite una comunicación inalámbrica en aplicaciones que posean fuentes de alimentación limitadas (p.e.: baterías) y de bajos requerimientos en cuanto a rendimiento. Los objetivos principales de una LR-WPAN son: facilidad de instalación, una transmisión de datos confiable, tener un relativo corto alcance de operación, muy bajo costo y ser capaz de mantener un protocolo sencillo y flexible manejando una durabilidad de batería(s) aceptable[9].

Algunas de las características que describen a una LR-WPAN son:

- Anchos de banda en aire de 250,40 y 20 Kbps<sup>1</sup>.
- Topologías de red punto-a-punto o estrella.
- Identificadores de red de 16 bits (cortos) o 64 bits (extendidos).
- Asignación de GTSs (Guaranteed Time Slots)<sup>2</sup>.
- Acceso a canal mediante mecanismos de acceso múltiple con evasión de colisión mediante detección de portadora o CSMA-CA, por sus siglas en inglés (Carrier Sense Multiple Acces - Colition Avoidance).
- Uso completo de protocolos de reconocimiento (Acknowledge) para alta confiabilidad en la transferencia de datos.
- Bajo consumo de potencia.
- Detección de energía en la entrada de la antena o ED (Energy Detection).
- Indicador de calidad de enlace o LQI (Link Quality Indication).
- Disponibilidad para 16 canales en la banda de los 2450MHz, 10 canales en la banda de los 915MHz y 1 canal para la banda de los 868MHz.

---

<sup>1</sup>Un kilobit por segundo es una unidad de medida que se usa en telecomunicaciones e informática para calcular la velocidad de transferencia de información a través de una red. Su abreviatura y forma más corriente es kbps o kb/s. El símbolo estándar internacional es kbit/s. Equivale a 1000 bits por segundo = 1000 bit/s.

<sup>2</sup>GTS se define como un bloque de tiempo reservado, durante el envío de un paquete, para la transmisión de información concerniente a aplicaciones que se ejecuten de forma paralela al funcionamiento de la red.

De igual forma, es necesario describir la jerarquía que se maneja dentro de una LR-WPAN para poder comprender sus aptitudes. Esta puede desglosarse en tres elementos básicos:

- Coordinador de red de área personal (PANCoordinator): es el encargado principal de la red, provee de mecanismos de sincronización mediante el uso de tramas de guías o beacons (descrita más adelante en este trabajo de grado), además de encargarse de los procesos de asociación, disociación, así como el reordenamiento y de nodos huérfanos<sup>1</sup>, entre otros. PANCoordinators también pueden ser utilizados para el enrutamiento de paquetes en la red y cada LR-WPAN puede tener sólo uno. Se caracterizan por tener una fuente de alimentación directa (no usan baterías) además de una mayor cantidad de recursos que los dispositivos finales (nodos comunes) y algunos coordinadores de la red (mayor capacidad de procesamiento, memoria RAM, etc). Se hará referencia al coordinador de red de área personal como PAN-C para facilitar el desarrollo del contenido.
- Coordinador (Coordinator): estos de igual forma dan soporte el sistema de sincronización de la red retransmitiendo las tramas de guías (beacons). Pueden contar con una fuente de alimentación directa o utilizar baterías. Además pueden o no disponer de la misma cantidad de recursos que un PAN-C pero siempre mayor cantidad que un dispositivo final. Existen dos tipos: los que pueden realizar labores como dicho PAN-C (coordinadores de red de área personal alternos) y los que no (simplemente "coordinadores").
  - Coordinadores de red de área personal alternos (alternate PAN-C): poseen las características fundamentales que requiere un coordinador de red (Alimentación directa, capacidad de procesamiento, capacidad para el manejo de procesos de asociación, entre otros). Es capaz de sustituir al actual coordinador de red en caso de ser necesario (salida de funcionamiento temporal o definitiva del PAN-C asignado) opera dentro de la red como un coordinador simple mientras su funcionamiento como PAN-C no sea necesario.
  - Coordinadores (Coordinators): no poseen las capacidades para suplir a un PAN-C, por lo que se ven limitados a los procesos de sincronización y enrutamiento aunque pueden realizar tareas de dispositivos finales.
- Dispositivo final (End Device): son el final de la rama de jerarquía y cuentan con una cantidad de recursos menor a la de los coordinadores ya que las tareas para las cuales están diseñados (medición, accionamientos, entre otros), no lo requieren.

Dentro de una LR-WPAN pueden existir dos tipos de dispositivos: dispositivos funciones reducidas o RFD (Reduced Function Devices) y los dispositivos de funciones extendidas o FFD (Full Function Devices). Un FFD puede operar como coordinador de la red de área personal (PAN-C), como coordinador (Coordinator) o como dispositivo final (End Device). Dichos FFDs pueden establecer comunicaciones con RFDs u otros FFDs, mientras que los RFDs sólo comunicarse con FFDs, además de sólo poder operar como dispositivos finales (End

---

<sup>1</sup>Un nodo huérfano se define como aquel nodo que ha perdido comunicación con su coordinador de red de área personal.[9]

Devices). Esto hace que los RFDs no tengan la necesidad de procesar grandes cantidades de información por lo que no requieren una gran cantidad de velocidad de procesamiento ni gran capacidad para almacenamiento de datos [9].

El estándar IEEE 802.15.4 está orientado a LR-WPANs, por lo tanto, regula todos los parámetros que hacen a una red ser una de esta índole.

### 2.2.2.1. Topologías

Las topologías de red pueden ser tanto estrella como punto-a-punto:

**Topología Estrella.** En la topología estrella, la comunicación se establece entre los dispositivos finales (pueden ser tanto FFD como RFD) y un dispositivo central denominado coordinador de red de área personal (PAN-C) como se muestra en la Figura 2.3. Dicho coordinador puede o no tener aplicaciones o tareas asignadas como los demás dispositivos pero debe ser capaz de enrutar cualquier tipo de información dentro de la PAN.

Todos los dispositivos deben tener asociado un identificador único de 64 bits y además se le puede ser asignado un identificador corto de 16 bits por el coordinador al momento de ser asociado a la PAN [9].

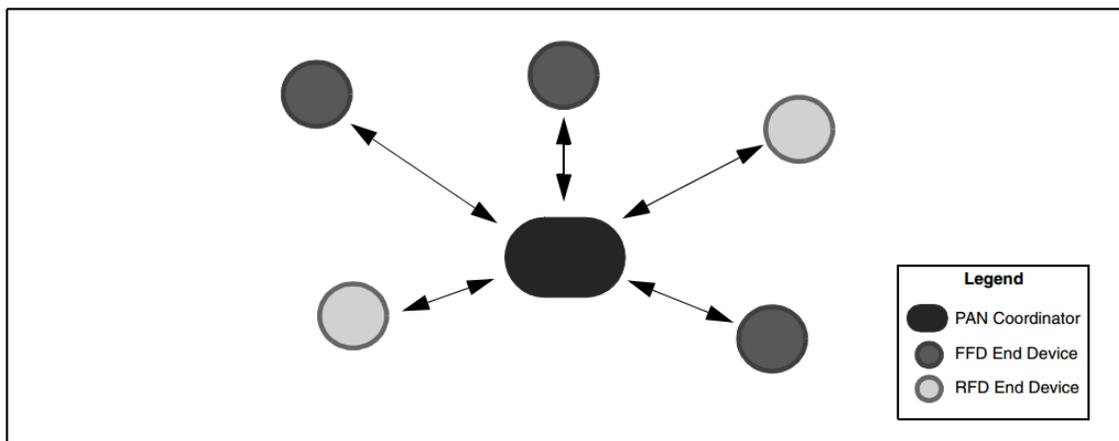


Figura 2.3: Ejemplo de topología tipo estrella.  
Referencia: [10]

**Topología Peer-to-Peer.** Al igual que la topología estrella, la topología peer-to-peer cuenta con un coordinador de red, pero a diferencia de la topología estrella, cada dispositivo de la red puede comunicarse con otro, siempre y cuando este último se encuentre dentro de su espacio personal de operación o POS, por sus siglas en inglés (Personal Operating Space). Un ejemplo de esta topología se muestra en la Figura 2.4 [9].

Al igual que la topología estrella se establecen identificadores de red para cada dispositivo, exactamente de la misma forma.

Esta topología admite otras aún más complejas como la topología mesh.

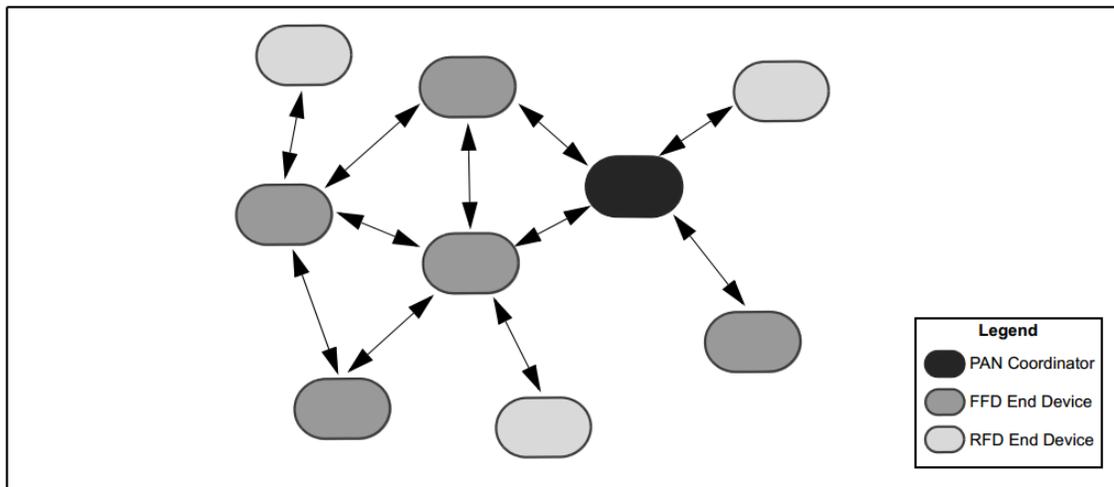


Figura 2.4: Ejemplo de topología peer-to-peer.  
Referencia: [10]

**Topología Mesh.** Dentro de la topología mesh, todos los dispositivos pertenecientes a la PAN pueden ser idénticos, salvo uno que debe servir como PAN-C. En este tipo de topología no existe una estructura predefinida como en la anteriormente mencionada topología estrella.

La información puede viajar de dispositivo en dispositivo hasta encontrar al nodo de destino, esto hace que el alcance de la red se vea extendido más allá del alcance propio de cada uno. Además, entre otros algoritmos permite el descubrimiento automático de nuevas rutas y se recupera de forma espontánea frente a fallos de la red (Salida de un nodo de la red, nodos huérfanos, entre otros).

La única restricción que se mantiene es que los RFDs siguen solamente pudiendo comunicarse con los FFDs[5].

#### 2.2.2.2. La Arquitectura

La arquitectura de las redes de tipo LR-WPAN está definida en función de bloques llamados capas, donde cada una es responsable por cierta parte del estándar además de prestar servicios a las capas superiores. La disposición de estas capas está basada en el modelo OSI.

Cualquier dispositivo que forme parte de una red basada en el estándar debe estar conformado por una capa física (PHY), la cual comprende todo lo concerniente al canal RF incluyendo mecanismos de control de bajos niveles de potencia, y una capa de control de acceso al medio (MAC) que provee el acceso al canal físico para la transferencia de cualquier paquete [9].

Las capas superiores constan de una capa de red, la cual proporciona el soporte para la configuración de la red en sí, su manipulación y enrutamiento de paquetes dentro de ella, y una capa de aplicación que provee de las funciones destinadas a cada dispositivo.

Cabe destacar que la definición de las capas superiores escapa del objetivo de este estándar ya que, como se mencionó anteriormente, este estándar sólo define los parámetros de la capa PHY y la MAC, sin embargo, fue diseñado un mecanismo de acceso a la MAC denominado "control de enlace lógico" o LLC (Logical Link Control) que accede a través de la subcapa

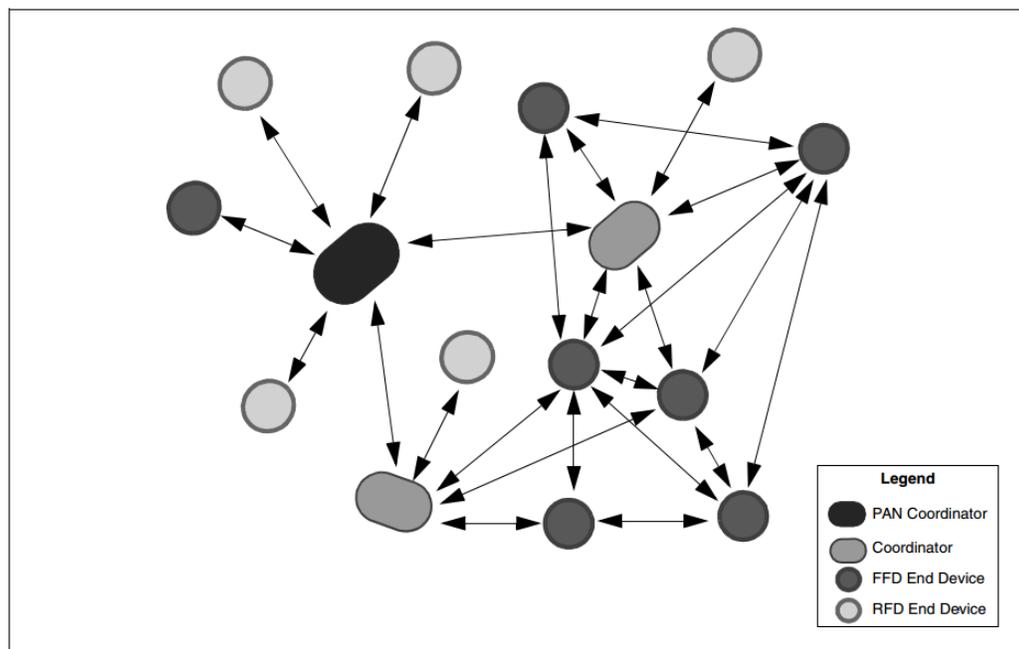


Figura 2.5: Ejemplo de topología mesh.  
Referencia: [11]

de servicio de convergencia específica o SSCS (Service Specific Convergence Sublayer) para de esta manera dejar, a disposición de quien lo desee, la implementación de dichas capas superiores.

Se puede observar en la Figura 2.6 la arquitectura descrita en esta pauta partiendo del canal físico o medio, las capas PHY, la MAC, etc. hasta llegar al bloque comprendido por el resto de las capas superiores.

### 2.2.2.3. Modos de Operación de una Red bajo el Estándar IEEE 802.15.4

Existen dos modos de operación: con y sin habilitación de guías o *beacons*. Dichos modos son denominados: *beacon enabled* (BE) y *nonbeacon enabled* (NBE) respectivamente.

**Modo Beacon Enabled.** En este modo todos los nodos, salvo el PAN-C, permanecen en un estado de bajo consumo durante un tiempo denominado "intervalo de *beacon*" o BI, por sus siglas en inglés (Beacon Interval), luego todos deben salir de dicho estado para la recepción del próximo *beacon*, el cual contiene información concerniente a paquetes pendientes por enviar del PAN-C y su destinatario. El nodo destino debe permanecer a la espera del paquete, mientras que el resto debe volver a dicho estado de bajo consumo. Luego de dicho nodo recibir el paquete, debe volver a su estado inicial, hasta el siguiente *beacon*. Esto hace que la red tenga un consumo potencia mucho menor.

El PAN-C es el encargado de emitir los anteriormente mencionados *beacons*, los cuales mantienen la sincronización de la red a razón del BI[9].

Este modo tiene como desventaja el tener que adoptar topologías estructuradas (estrella, árbol, entre otras) debido a que, al ser el PAN-C el que sincroniza la red, solo él maneja la

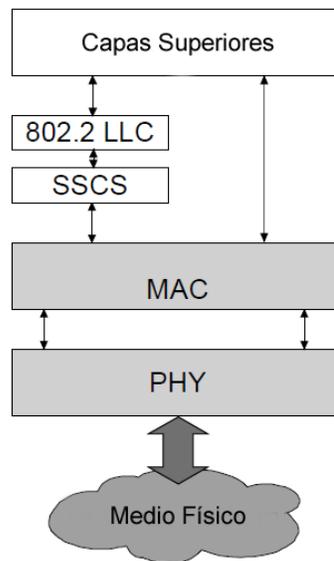


Figura 2.6: Arquitectura del Estándar IEEE 802.15.4.  
Referencia: [9]

información transmitida, imposibilitado comunicaciones de dispositivo a dispositivo (tipo de comunicación base para la topología mesh).

**La Supertrama (Superframe).** Dentro del modo de operación *beacon enable*, existe una modalidad denominada Supertrama o Superframe. Esta consiste en la segmentación del BI en dos regiones básicas, como se observa en la Figura 2.7, denominadas porción activa (Active Portion), cuya duración se denomina "duración de supertrama" o SD (Superframe Duration) y la otra, porción inactiva (Inactive Portion). La primera, a su vez, se desglosa en dieciséis(16) bloques (Slots), para el envío de información. Estos pueden ser ocupados en su totalidad o no, durante la transmisión de un paquete, lo que implica un ancho de banda variable. Algunas aplicaciones requieren la regulación de dicho ancho de banda por lo que se limita el tiempo de transmisión de información a un intervalo denominado "tiempo de contención para acceso" o CAP (Contention Access Period), dejando dentro de la porción activa un tramo denominado "tiempo de contención libre" o CFP (Contention Free Period). Este último puede tener una duración variable de cero(0) a siete(7) slots. Cabe destacar que algunas aplicaciones utilizan el CFP para la transmisión de información concerniente a aplicaciones que se ejecutan paralelamente al funcionamiento de la red. Dicha información se transmite mediante el uso de "bloques de tiempo garantizados" o GTSs (Guaranteed Time Slots), los cuales pueden ocupar más de un Slot.

Se puede observar en la Figura 2.7, la estructura de la supertrama, la cual inicia su CAP (en este caso de 9 slots) con la recepción del primer *beacon* (este *beacon* está contenido dentro del slot cero). Inmediatamente después de concluir el CAP sigue el CFP. Se puede observar en dicha Figura como el CFP está compuesto por tres(3) GTSs (GTS1 de tres slots, GTS2 de un slot y GTS3 de tres slots para un total de siete slots, el máximo permitido). Luego con el fin del CFP culmina la porción activa e inicia la inactiva donde se regresa al estado inicial

de bajo consumo hasta la recepción del próximo *beacon*.

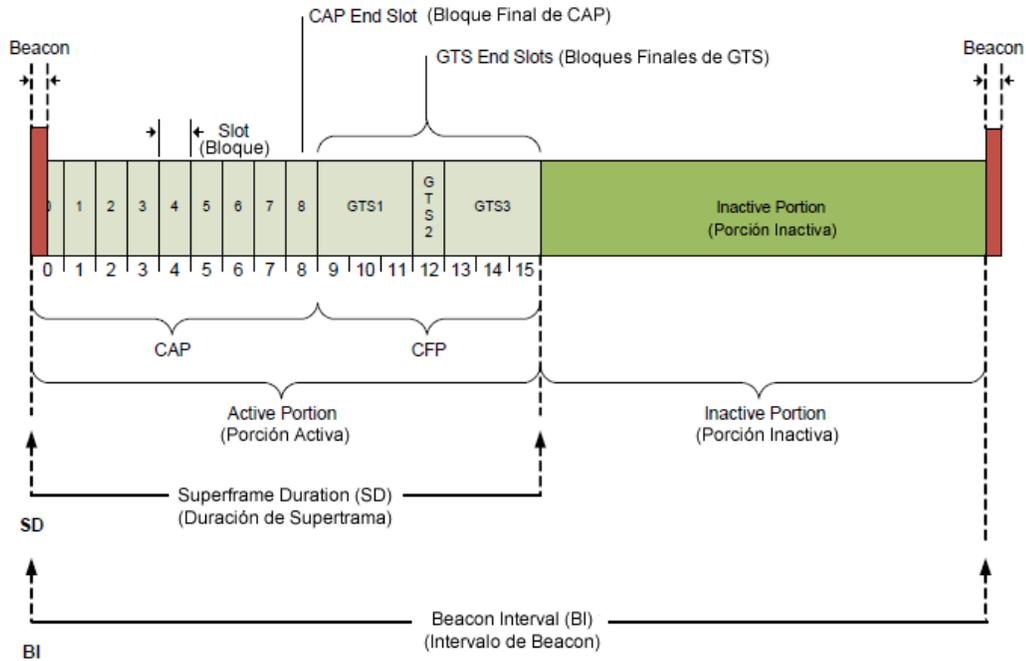


Figura 2.7: Estructura de Supertrama.  
Referencia: [9]

**Modo Nonbeacon Enabled.** Este modo, aunque no es totalmente carente de *beacons*, por ser estos necesarios para el proceso de asociación, permite una red con menos uso del canal ya que no implementa las tramas de guía para ello, tramas que en el modo *beacon enable* son transmitidas cada BI a pesar de no ser necesario el envío de información. Este hecho representa un ahorro de energía parcial para este modo de operación ya que no se envían datos de no ser estrictamente necesario. Como desventaja de este, es el hacer obligatoria la continua atención al canal lo que le imposibilita, a los nodos de la red, del uso de mecanismos de ahorro de energía, aumentando así el consumo generalizado de la red.

**Modelo de Transferencia de Datos.** Este modelo implica tres(3) tipos de transferencias: coordinador a dispositivo, dispositivo a coordinador y dispositivo a dispositivo. Dos de éstas son usadas en la topología estrella mientras que la topología peer-to-peer hace uso de los tres tipos.

Dependiendo del modo bajo el que opere la red, se tendrán esquemas de funcionamiento como los mostrados en las Figuras 2.8 y 2.9.

#### 2.2.2.4. Descripción de la Capa PHY

La capa física o PHY (nomenclatura del estándar), como fue mencionado en la descripción del modelo OSI, es la que está encargada de regular todos los parámetros en cuanto al medio. Como se puede observar en el Cuadro 2.1 la capa física para el estándar IEEE 802.15.4 define

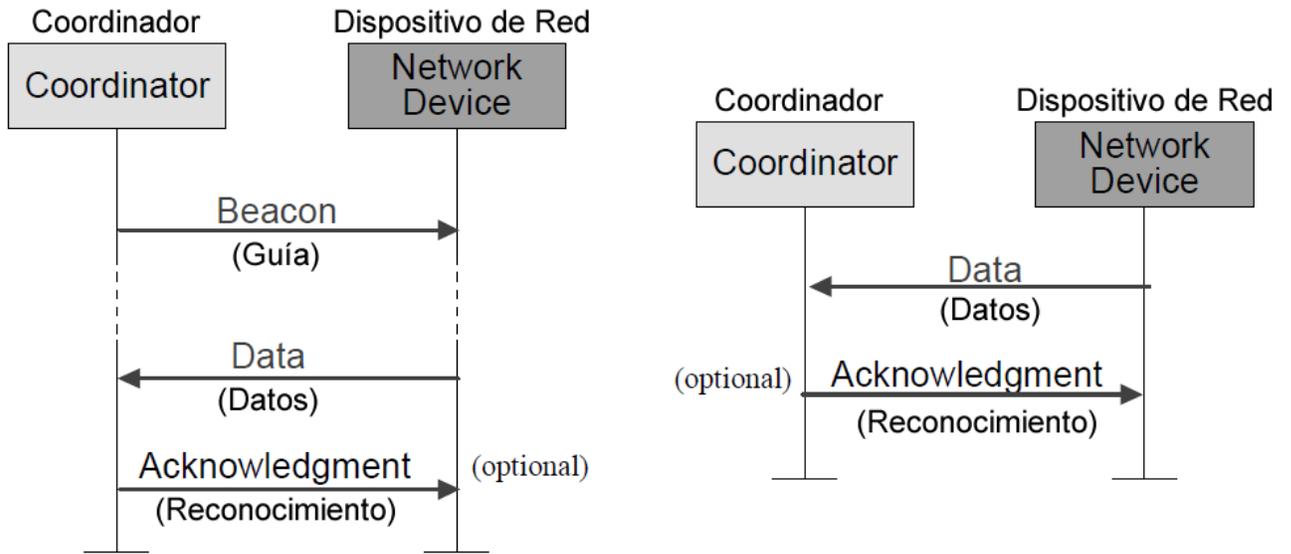


Figura 2.8: Comunicación dispositivo a coordinador. A la izquierda funcionamiento bajo modo *beacon enable* a la derecha bajo modo *nonbeacon enable*.

Referencia: [9]

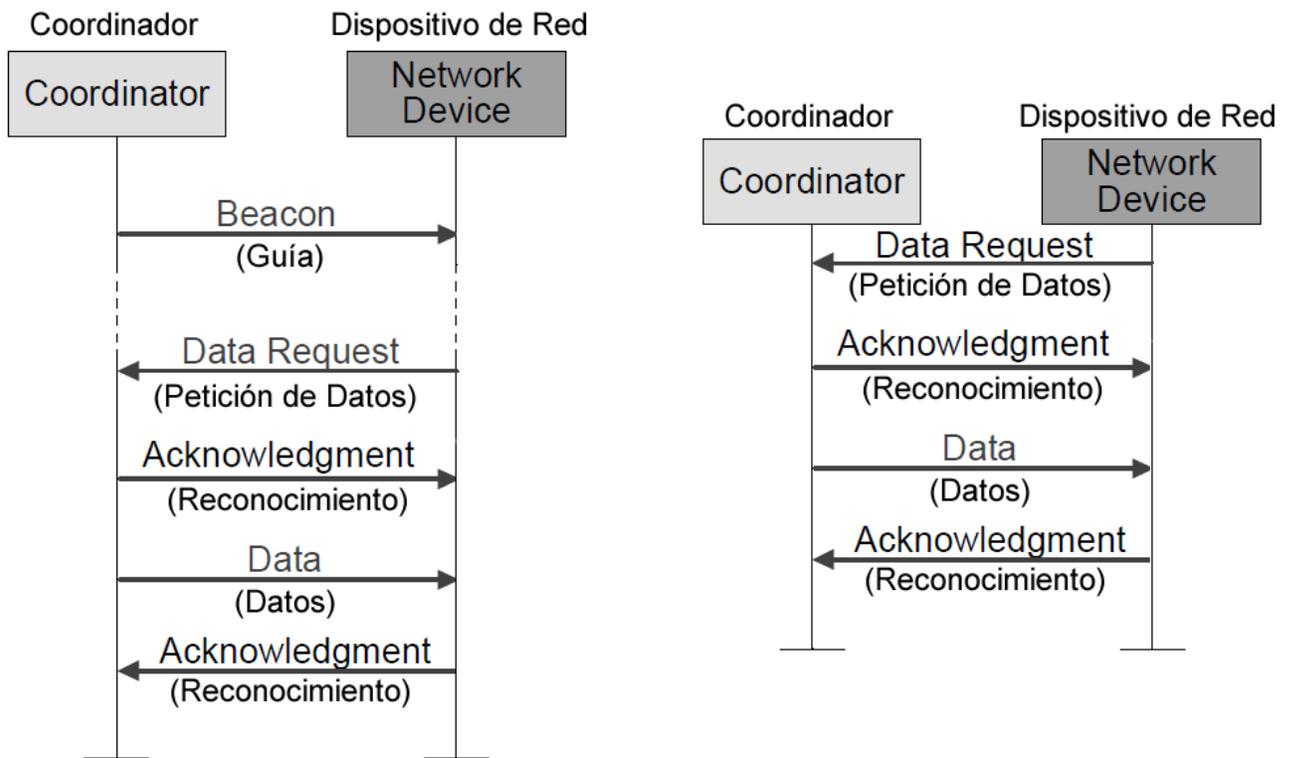


Figura 2.9: Comunicación coordinador a dispositivo. A la izquierda funcionamiento bajo modo *beacon enable* a la derecha bajo modo *nonbeacon enable*.

Referencia: [9]

rangos de operación en las bandas de 868, 915 y 2450 MHz; una tasa de pulsos de 300, 600 y 2000 Kchip/s<sup>1</sup>; define las modulaciones a implementar como BPSK para las bandas de 868 y 915MHz y O-QPSK para la banda de 2.4GHz; además regula el ancho de banda para 20, 40 y 250 Kbps; entre otros parámetros.

Cuadro 2.1: Algunos parámetros concernientes a la capa física (PHY) perteneciente al estándar 802.15.4.

Referencia: [9]

PHY (MHz)	Bandas de Frecuencia (MHz)	Parámetros de propagación		Parámetros de la Información		
		Tasa de Chip (Kchip/s)	Modulación	Ancho de Banda (Kb/s)	Tasa de Símbolo (ksymbol/s)	Symbols
868/915	868–868.6	300	BPSK	20	20	Binary
	902–928	600	BPSK	40	40	Binary
2450	2400–2483.5	2000	O-QPSK	250	62.5	16-ary Orthogonal

La capa física define como medio el aire e incluye mecanismos de ED, LQI, CCA, el algoritmo CSMA-CA, la selección de canal y habilita tanto la recepción como emisión de paquetes.

### Algunos Mecanismos del Estándar IEEE 802.15.4.

- a) **Detección de energía (ED)**. Es un mecanismo pensado para ser utilizado por la capa de red como parte de un algoritmo de selección de canal. Provee un valor estimado de la potencia de la señal medida en el canal actual. Esta medición es presentada mediante un valor entero de 8 bits contenido dentro del rango 0x00 - 0xFF hexadecimal, donde el mínimo valor corresponde con el 0x00, representando una potencia menor a 10dB por encima de la sensibilidad especificada para el receptor, y donde el máximo corresponde a 40dB por encima de la referencia de 10dB antes mencionada [12].
- b) **Indicador de calidad de enlace (LQI)**. Es un mecanismo que caracteriza la fuerza y/o la calidad de un paquete recibido. Su cálculo utiliza el valor de energía medido por el mecanismo ED en el receptor, un estimado de la relación señal-a-ruido o una combinación de ambos. Este valor está contenido dentro del rango 0x00 y 0xFF hexadecimal, correspondiendo 0x00 con el valor mínimo y 0xFF con el máximo. Cabe destacar que la capa de implementación de este mecanismo (capa de red o la capa de aplicación), no está especificada dentro del estándar [12].
- c) **Referencia de canal libre (CCA)**. Dentro del algoritmo de detección de canal libre existen algunos parámetros que deben ser prefijados según se desea la forma de operación de la red. Dentro de estos se encuentra el modo de detección de canal libre. Dicho modo permite ajustar bajo cuales circunstancia se debe reportar el canal como ocupado o libre [12]. Estos modos se listan a continuación:

<sup>1</sup>Un chip/s equivale a un bit/s de la señal radiada.

Octets: 4	1	1		variable
Preamble	SFD	Frame length (7 bits)	Reserved (1 bit)	PSDU
SHR		PHR		PHY payload

Figura 2.10: Composición de la capa física (PHY).  
Referencia: [9]

- o Modo 1 - Energía por encima del umbral: el mecanismo CCA debe reportar el canal como ocupado al detectar energía por encima del umbral del ED.
  - o Modo 2 - Sólo detección de portadora: el mecanismo de CCA debe reportar el canal como ocupado cuando se detecte una señal con las características de modulación descritas en el estándar IEEE 802.15.4, esto sin importar que los niveles de energía medidos, se encuentren por encima o por debajo del umbral de detección del mecanismo ED.
  - o Modo 3 - Detección de portadora con energía por encima del umbral: como combinación de los dos modos descritos previamente, dentro de este modo de operación el mecanismo CCA debe reportar el canal como ocupado cuando se detecte una señal con las características de modulación descritas dentro del estándar y la energía medida sea mayor al umbral de detección del mecanismo ED.
- d) **El mecanismo CSMA-CA.** Este algoritmo permite el acceso al canal, haciendo uso de los mecanismos de CCA y ED además de un conjunto de parámetros internos del estándar. Da soporte a dos mecanismos el CSMA-CA Ranurado y el CSMA-CA No Ranurado para las configuraciones *beacon enabled* y *nonbeacon enabled* respectivamente. Tanto el algoritmo CSMA-CA como dichas configuraciones serán tratadas más adelante dentro de este trabajo de grado.

**La Unidad de Datos Protocolar de la Capa PHY (PPDU).** El estándar plantea un formato para la unidad de datos manejada por la capa física que consta de tres componentes básicos como lo son: el encabezado de sincronización o SHR (Synchronization Header), el encabezado de la capa física o PHR (PHY Header) y una trama de carga o *payload* de longitud variable que soporta a la capa MAC. En la Figura 2.10 se muestran las componentes de la PPDU y los campos que la componen los cuales se describen a continuación:

- a) **Preámbulo o Preamble.** Este campo es usado por los transceptores para sincronización. Consta de 32 ceros en binario y permite la sincronización de los chip/s (pulsos por segundo) y los symbol/s (símbolos por segundo) a la hora de recibir un paquete.
- b) **Delimitador de Inicio de Trama o SFD.** Es un campo de 8 bits y actúa como indicador de fin del campo *preamble*, o en otras palabras, el fin de la sincronización y por ende el comienzo del paquete.
- c) **Longitud de Trama.** Este campo está conformado por 7 bits y muestra la longitud del campo PSDU. Este valor puede ir de 0 a 127 octetos. Cabe destacar que dentro del

Octets: 2	1	0/2	0/2/8	0/2	0/2/8	variable	2
Frame control	Sequence number	Destination PAN identifier	Destination address	Source PAN identifier	Source address	Frame payload	FCS
		Addressing fields					
MHR						MAC payload	MFR

Figura 2.11: Estructura básica de la capa MAC.  
Referencia: [9]

estándar el número máximo de octetos se denomina a `MaxPHYPacketSize` y su valor es 127.

**Unidad de Datos de Servicio de la Capa PHY (PSDU).** Es un campo de longitud variable el cual tiene como carga los paquetes que forman la información a transportar. Dentro de este campo está contenida la capa MAC.

#### 2.2.2.5. Descripción de la Capa MAC

La capa MAC es la que maneja todos los accesos al canal RF de comunicaciones y además es responsable de un conjunto de tareas listada a continuación:

- Generación de guías o *beacons* si el dispositivo es un coordinador.
- Establecer el sincronismo con los *beacons*.
- Ofrecer soporte para dispositivos que deseen asociarse a la PAN.
- Ofrecer el soporte para la seguridad al dispositivo.
- Emplear el mecanismo CSMA-CA para el acceder al canal.
- Manejo y mantenimiento del mecanismo GTS.
- Proveer un enlace confiable entre entidades MAC.

**La Unidad de Datos Protocolar de la Capa MAC (MPDU).** La capa MAC, al igual que la capa PHY, está formada por tres componentes los cuales son: el encabezado o MHR (Mac Header), la carga o MAC *payload* y por último el pie de trama o MFR (Mac Footer). En la Figura 2.11 se observa la estructura de la MAC mencionada anteriormente además de los campos que internamente la componen, los cuales serán descritos a continuación:

- a) **Control de trama (frame control).** Campo de dos octetos de longitud, el cual contiene información que define el tipo de trama que se está recibiendo. Los tipos de identificadores que vienen contenidos en el paquete (16bits o 64bits), entre otros. El formato de este campo corresponde con el observado en la Figura 2.12.

Se describirán a continuación los campos pertenecientes al frame control:

Bits: 0–2	3	4	5	6	7–9	10–11	12–13	14–15
Frame type	Security enabled	Frame pending	Ack. request	Intra-PAN	Reserved	Dest. addressing mode	Reserved	Source addressing mode

Figura 2.12: Estructura del campo control de trama (*frame control*).  
Referencia: [9]

- **El tipo de trama (Frame Type)**. Subcampo de 3 bits que indica qué tipo de trama es la que está siendo recibida, ya sea *beacon*, *data*, *acknowledgement* o *command*. Los valores que toma este subcampo corresponden con la Cuadro 2.2. Estos tipos de tramas serán descritos más adelante dentro de este trabajo de grado.

Cuadro 2.2: Valores del subcampo tipo de trama (*frame type*).  
Referencia: [9]

Frame type value $b_2 b_1 b_0$	Description
000	Beacon
001	Data
010	Acknowledgment
011	MAC command
100–111	Reserved

- **Habilitación de seguridad (Security Enable)**. Subcampo de 1 bit de longitud y muestra si el paquete recibido se encuentra encriptado o no, para los valores de cero y uno respectivamente.
- **Trama pendiente (Frame Pending)**. Subcampo de un 1 bit de longitud que indica, al ser su valor igual a uno(1), que el dispositivo emisor tiene paquetes pendientes por transmitir a este mismo destino, de lo contrario este subcampo debe permanecer en cero(0).
- **Solicitud de verificación de correcta recepción (Ack Request)**. Subcampo de un 1 bit de longitud que indica, al ser su valor igual a uno(1), que el dispositivo emisor solicita se le sea enviado un paquete tipo *acknowledge*. En caso contrario este subcampo debe permanecer en cero(0).
- **Intra PAN**. Subcampo de 1 bit de longitud que indica, si el valor que toma es igual a uno(1), que el paquete recibido tiene como destino una dirección interna dentro de la PAN.
- **Tipo de identificador de destino (Destination Addressing Mode)**. Subcampo de 2 bits de longitud que muestra, según el Cuadro 2.3, qué tipo de identificadores de destinos están contenidos dentro del paquete que está siendo recibido.

Cuadro 2.3: Valores posibles para los subcampos *destination addressing mode* y *source addressing mode* del campo *frame control*.

Referencia: [9]

Addressing mode value $b_1 b_0$	Description
00	PAN identifier and address field are not present.
01	Reserved.
10	Address field contains a 16 bit short address.
11	Address field contains a 64 bit extended address.

- o **Tipo de identificador de origen (Source Addressing Mode)**. Subcampo de 2 bits de longitud que muestra, según el Cuadro 2.3, qué tipo de identificadores de origen están contenidos dentro del paquete que está siendo recibido.
- b) **Número de Secuencia (Sequence Number)**. Campo de 8 bits que permite determinar el orden de los paquetes recibidos, ya que debe incrementarse en pasos de uno en uno con la transmisión de cada uno. El valor que este tome al iniciar cada dispositivo debe ser generado bajo un algoritmo aleatorio cuyo desarrollo escapa de este estándar.
- c) **Identificador de PAN de Destino (Destination PAN Identifier)**. Campo de longitud variable (entre cero y dos octetos), que puede o no contener el identificador corto de 16 bits como identificador de PAN de destino, en caso de ser necesario.
- d) **Identificador de Destino (Destination Address)**. Campo de longitud variable (entre cero, dos u ocho octetos) que puede o no contener, o un identificador corto de 16 bits o extendido de 64 bits como destino, de ser necesario.
- e) **Identificador de PAN de Origen (Source PAN Identifier)**. Campo de longitud variable (entre cero y dos octetos), que puede o no contener un identificador corto de 16 bits como identificador de PAN de origen, de ser necesario.
- f) **Identificador de Origen (Source Identifier)**. Campo de longitud variable (entre cero, dos u ocho octetos), según indique el subcampo *destination addressing mode* del *frame control*, que pueden contener o no, un identificador corto o uno extendido de 64 bits como identificadores de origen, de ser necesario.
- g) **Carga de la Trama (Frame Payload)**. Es un campo de longitud variable que contiene la información enviada. Aquí se encuentran contenidas la capa de red y la de aplicación.
- h) **Trama de Chequeo de Secuencia (FCS)**. Es un campo de 16 bits de longitud que contiene el CRC (Cyclic Redundancy Check) de 16 bits de la ITU-T <sup>1</sup>.

<sup>1</sup>La ITU (International Telecommunication Union) fue creada en el año 1865. Se encarga de coordinar los principales estándares a nivel de telecomunicaciones del mundo. Su extensión en telecomunicaciones (ITU-T) fue creada en el año 1956 con el nombre de CCITT, por sus siglas en francés, Comité Consultatif International Téléphonique et Télégraphique, renombrada en 1993 para adoptar el nombre que hoy lleva.

**Tipos de Tramas** Existen un conjunto de tramas que son manejadas dentro del estándar, las cuales tienen propósitos específicos. Estas se describen a continuación:

- a) **Trama de Reconocimiento (Acknowledge Frame)**. Esta es una trama cuyo propósito es la notificación de la correcta recepción de un paquete cualquiera, durante un proceso de comunicación. El envío de esta trama debe ser solicitado por el emisor del paquete, poniendo en alto el subcampo *acknowledge request* del campo *frame control* de la capa MAC. Se observa en la Figura 2.13 que esta trama carece, a nivel de la capa MAC, de su componente de carga o MAC *payload* o MSDU.

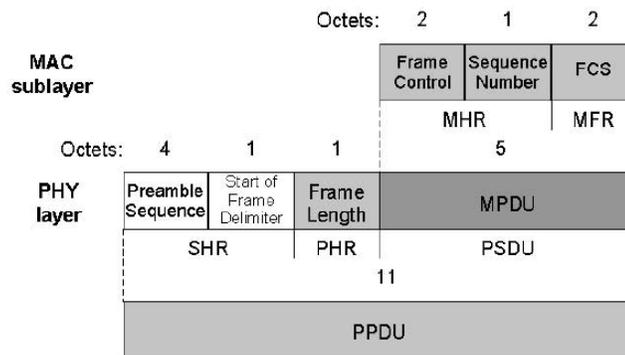


Figura 2.13: Estructura de un paquete de reconocimiento o *acknowledge*. Referencia: [9]

- b) **Trama de Guía (Beacon Frame)**. Esta trama es utilizada tanto para el proceso de asociación a la red como en el proceso de comunicación (sólo redes que funcionen bajo el modo *beacon enable*). Posee especificaciones del modo de operación contenidas en los campos *superframe specification* y *GTS fields* pertenecientes a la componente MSDU como se observa en la Figura 2.14.

A diferencia de la trama de *acknowledge*, la trama *beacon*, sí posee la componente de carga de la MAC y a la vez una componente de carga específica del paquete denominada como *beacon payload*.

Se puede observar en la Figura 2.14 su composición a nivel de componentes, capas y campos que la conforman.

- c) **Trama de Comandos (Command Frame)**. Esta trama es utilizada para todos los procesos de la red como entre algunos se pueden mencionar: solicitudes de asociación, disociación, petición de datos, notificaciones de conflicto, entre otros. Dentro de su componente MSDU como primer octeto tiene el campo *command type*, como se muestra en la Figura 2.15, cuyos valores a tomar se rigen por el Cuadro 2.4 según sea la naturaleza del comando. Algunos de dichos comando serán descritos con mayor detenimiento en la sección 2.2.2.6.

Cabe destacar, que se debe configurar correctamente el subcampo *frame type* del *frame control* para marcar mostrar al receptor el tipo de paquete que se está enviando. La

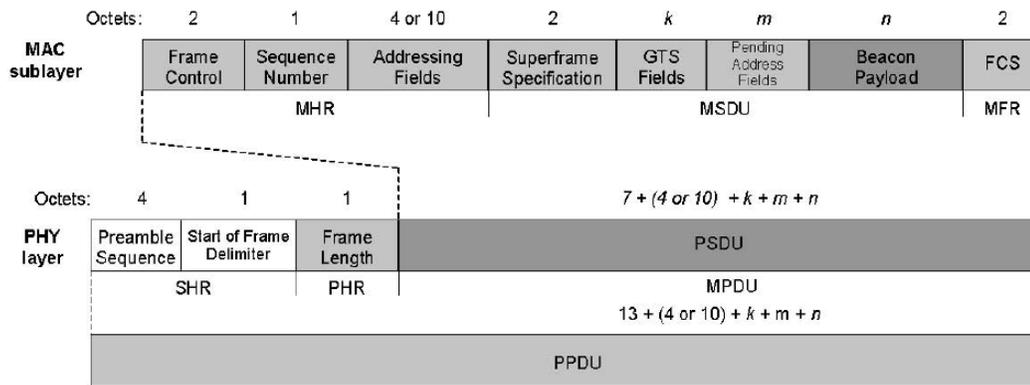


Figura 2.14: Estructura de una trama de guía o *beacon*. Referencia: [9]

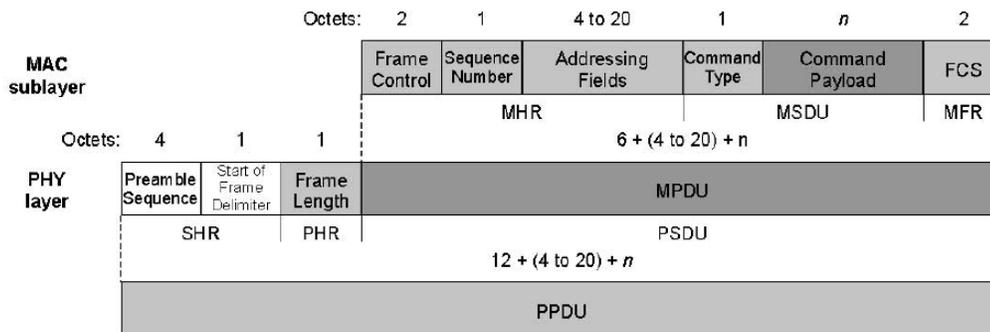


Figura 2.15: Estructura de un paquete de comando o *command*. Referencia: [9]

implementación del *beacon* como parte del proceso de asociación, será mostrada más adelante en la sección 2.2.2.6.

- d) **Trama de Datos (Data Frame).** Esta trama es la encargada de transportar la información dentro de su MSDU. Su estructura es básicamente la misma de la MAC corriente, como se observa en la Figura 2.16, esto debido a que no posee ningún campo dentro de la componente MSDU que la diferencie en gran medida. Al igual que tramas anteriormente descritas es necesaria la correcta configuración de los campos pertinentes como por ejemplo: especificar paquete encriptado, los tipos de identificadores se encuentran contenidos dentro del paquete, entre otros.

### 2.2.2.6. El Proceso de Asociación de un Nodo a la Red (Modo Nonbeacon Enable)

Es un proceso que, como su nombre lo indica, le permite a un nodo cualquiera asociarse a la red. Cabe destacar que se desarrollará únicamente el proceso de asociación para una red operando bajo el modo *nonbeacon enable*, debido a que este es el modo implementado para

Cuadro 2.4: Comandos asociados a la trama *command*.  
Referencia: [9]

Command frame identifier	Command name (Nombre de comando)
0 x 01	Association request (Petición de asociación)
0 x 02	Association response (Respuesta a la petición de asociación)
0 x 03	Disassociation notification (Notificación de disociación)
0 x 04	Data request (Petición de datos)
0 x 05	PAN ID conflict notification (Notificación de conflicto de PAN ID)
0 x 06	Orphan notification (Notificación de nodo huérfano)
0 x 07	Beacon request (Petición de trama de guía)
0 x 08	Coordinator realignment (Reordenamiento de nodo huérfano)
0 x 09	GTS request (Petición de GTS)
0 x 0a—0 x ff	Reserved (Reservado)

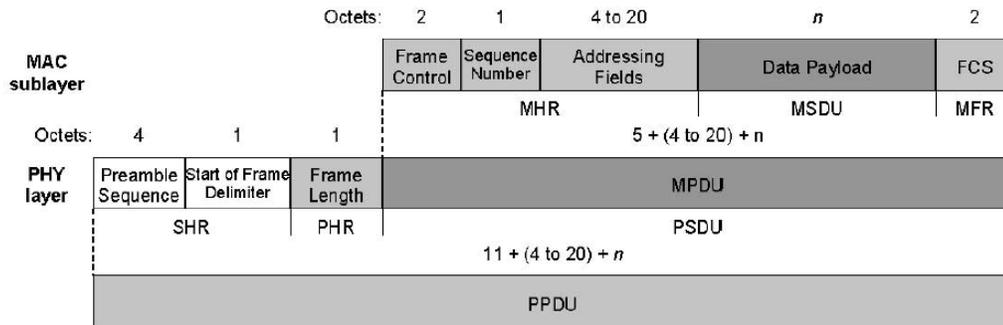


Figura 2.16: Estructura de un paquete de datos o *data*.  
Referencia: [9]

la red desarrollada en este trabajo especial de grado<sup>1</sup>.

A continuación será descrito el proceso de asociación en sus dos presentaciones: la asociación de un dispositivo que se encuentra dentro del alcance del PAN-C y la asociación de un dispositivo fuera de él. Este último será descrito en la sección 2.2.3.

**Asociación Dispositivo - coordinador** En este proceso, el dispositivo que desea asociarse (un nuevo coordinador o dispositivo final), se encuentra dentro del alcance del PAN-C por lo que el uso de las capas superiores (capa de red y aplicación) no se hace necesario. Esto permite esquematizar de una manera más simple dicho proceso como se muestra en la Figura 2.17.

Consta de 6 etapas básicas como lo son: la petición de trama de guía o *beacon request*, el *beacon* per sé como respuesta a dicha petición, la petición de asociación o *association request*,

<sup>1</sup>Se deja a criterio del lector, de interesarle, la revisión de la referencia [9] para la descripción del proceso de asociación de un nodo a una red operando bajo el modo *beacon enable*

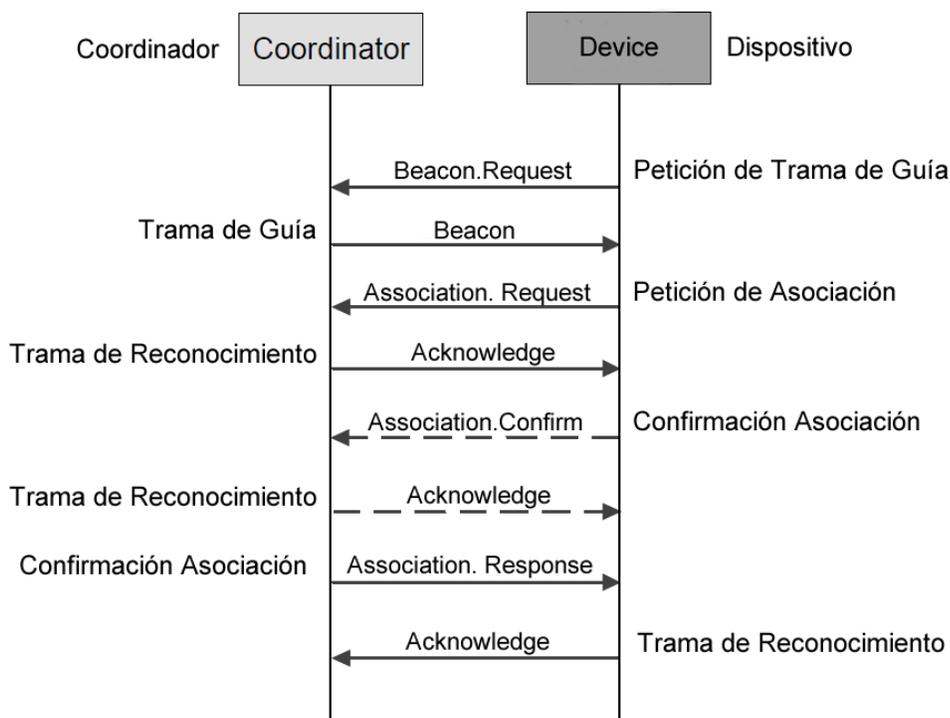


Figura 2.17: Esquema del proceso de asociación de un nodo a la red funcionando bajo el modo *nonbeacon enable*.

la confirmación de asociación o *association confirm*, la respuesta a la petición de asociación o *association response* y por último el *acknowledge*. Estas etapas serán descritas a continuación:

- a) **Petición de Beacon (Beacon.Request)**. Para un dispositivo poder asociarse (un nuevo coordinador o dispositivo final), debe averiguar primeramente si existe alguna red ocupando el canal en el que este se encuentre; y de existir, debe a su vez poder saber si dicha red posee disponibilidad para asociarlo. Esto se logra mediante el *beacon request*, ya que al ser esta una petición, su respuesta indicaría que el canal se encuentra ocupado por una red desarrollada bajo el estándar IEEE 802.15.4, y a su vez, le permite a dicho nodo, obtener el modo de operación de dicha red y todas sus características, como por ejemplo (identificador de PAN, disponibilidad por parte de la red para nuevas asociaciones, entre otros).

El *beacon request*, es transmitido mediante una trama de comando, tipo de trama en la que debe ser indicado en el subcampo *frame type* del *frame control*. Dicha trama de comando lleva en su campo *command frame identifier*, el valor 0x07 hexadecimal según el Cuadro 2.4, correspondiente a una petición de trama de guía. El *beacon request* porta, tanto para el identificador de PAN de destino, como para el identificador de dispositivo de destino (identificadores cortos de 16 bits), el valor 0xFFFF hexadecimal, lo que indica que va dirigido a cualquier dispositivo perteneciente a cualquier PAN disponible. Esta trama no porta identificadores de origen de ningún tipo. Se recuerda que el tipo de identificadores contenidos dentro de cada paquete debe ser especificado dentro de

octets: 7	1
MHR fields	Command frame identifier

Figura 2.18: Formato de Comando tipo *beacon request*.  
Referencia: [9]

los subcampos *destination addressing mode* y *source addressing mode* pertenecientes al campo *frame control* de la capa MAC.

Si un *beacon request*, no recibe respuesta alguna, el dispositivo en busca de PAN, debe asumir que dentro del canal en el que se encuentra está libre, por lo que debe realizar un cambio de dicho canal y realizar nuevamente dicha petición. Esta situación debe mantenerse hasta que el dispositivo logre asociarse a alguna red disponible.

En la Figura 2.18 se muestra el formato de un comando tipo *beacon request*.

- b) **Respuesta a la Petición de Beacon.** Como respuesta a un *beacon request*, por parte del PAN-C, es enviado un *beacon*. Este posee la información que describe todo el modo de operación de la red y sus características. Uno de los campos que conforman al *beacon*, es el *superframe specifications*, dentro del cual se encuentra el subcampo *association permit* (un bit de longitud), el cual indica, de estar en estado alto, que la PAN se encuentra admitiendo nuevos dispositivos a la red. A su vez, dicho *beacon*, le brinda información al dispositivo del identificador de PAN y del identificador del coordinador de la red que emisor de la respuesta.

Al dispositivo recibir el *beacon*, y detectar que se están aceptando nuevas asociaciones, debe emitir un *association request*, de lo contrario, continuar con la búsqueda de una PAN a la cual asociarse.

- c) **Petición de Asociación (Association.Request).** Al igual que el *beacon request*, el *association request* se transmite bajo la trama de tipo comando, con un *command frame identifier* = 0x01 hexadecimal, correspondiente a un *association request*. Esta trama contiene como identificador de PAN de destino e identificador de coordinador de destino, los obtenidos previamente del *beacon* (respuesta al *beacon request*). A su vez, toma el valor 0xFFFF, como identificador de PAN de origen, indicando el no estar asociado a ninguna red. Además contiene el identificador extendido de 64bits único de cada dispositivo en el mundo<sup>1</sup>. Como primer byte, del *command payload*, se encuentran descritas, en un octeto, las aptitudes propias del dispositivo para con la red (*capability information*), dentro de las cuales se pueden mencionar: el tipo de dispositivo (RFD o

<sup>1</sup>Los veinticuatro(24) bits más significativos del MAC *address* (identificador extendido de 64 bits), se denominan OUI (Organizational Unique Identifier), el cual debe ser comprado a la IEEE, ente que regula su distribución para evitar conflictos entre los nodos de cualquier red en el mundo o incluso nodos de redes vecinas. El resto de los bits se denominan OEM (Original Equipment Manufacturer) y son regulados directamente por el consumidor intermedio (desarrolladores de dispositivos basados en este estándar) [13] [14].

octets: 17/23	1	1
MHR fields	Command frame identifier	Capability information

Figura 2.19: Formato de la petición de asociación.  
Referencia: [9]

FFD), tipo de fuente de alimentación (directa o baterías), capacidad para manejo de encriptación y desencriptación de información, entre otros.

Además de todo lo antes mencionado, la petición de asociación debe estar respaldada por mecanismos de *acknowledge*, por lo tanto, dentro de su encabezado se debe configurar en alto el subcampo *Ack. Req* del *frame control* para indicarle al coordinador que debe notificar la correcta recepción de este paquete. En la Figura 2.19 se observa el formato de un comando tipo *association request*.

- d) **Confirmación de Asociación (Association.Confirm)**. Existe la posibilidad de que por alguna u otra circunstancia, la petición de asociación no sea respondida, en este caso, el dispositivo que desea asociarse, debe emitir una confirmación de asociación (*association confirm*).

Dicha petición es de tipo comando, por lo tanto, se envía bajo la trama *command frame*. El campo *command frame identifier* se escribe con 0x04 hexadecimal (*data request*), correspondiente con el Cuadro 2.4, por ser una petición de datos.

Esta confirmación debe contener como identificador de PAN de origen, el identificador de PAN de la red a la que se desea asociar y el identificador extendido de origen de 64 bits. A diferencia del *association request*, el *association confirm* no posee ningún tipo de carga. Este comando requiere de igual forma el uso de mecanismos de *acknowledge* para verificar su correcta recepción por parte del coordinador. En la Figura 2.20, se observa el formato de la confirmación de asociación.

octets: 7/11/13/17	1
MHR fields	Command frame identifier

Figura 2.20: Formato de la confirmación de asociación (Data Request).  
Referencia: [9]

- e) **Respuesta a la Petición de Asociación (Association.Response)**. Finalmente, la respuesta a la petición de asociación, la cual puede observarse en la Figura 2.21, es la que concluye el proceso, sin importar si el nodo fue aceptado o no por el PAN-C.

Este paquete se transmite bajo la trama *command frame* con *command frame identifier* = 0x02 hexadecimal, correspondiente a un comando tipo *association response* según el Cuadro 2.4. Este finalmente contiene la información del estado de la asociación,

octets: 23	1	2	1
MHR fields	Command frame identifier	Short address	Association status

Figura 2.21: Formato de la respuesta a la petición de asociación (*association response*). Referencia: [9]

Cuadro 2.5: Estados de asociación posibles para el campo *status*. Referencia: [9]

Association status (Estado de Asociación)	Description (Descripción)
0 x 00	Association successful. (Asociación satisfactoria)
0 x 01	PAN at capacity. (PAN a plena capacidad)
0 x 02	PAN access denied. (Acceso a PAN denegado)
0 x 03—0 x 7f	Reserved. (Reservado)
0 x 80—0 x ff	Reserved for MAC primitive enumeration (Reservado)

indicándolo en el campo *association status* del *command payload* según el Cuadro 2.5, donde se contemplan: una asociación positiva para un *status* = 0x00 hexadecimal, una negativa para un *status* = 0x01, y finalmente la posibilidad de serle negado el acceso a la PAN al dispositivo *mediante un status* ajustado en 0x02 (red a capacidad, limitación del número de nodos de la red, entre otros). En este último caso, el dispositivo debe mantenerse en el proceso de buscar una PAN a la cual asociarse.

Se puede observar en la Figura 2.18, un campo denominado *short address* en el cual, de haberse solicitado por parte del dispositivo, el Identificador Corto de 16 bits.

Esta etapa de igual forma debe ser respaldada por mecanismos de *acknowledge*.

### 2.2.3. Las Capas Superiores

En vista del soporte que ofrece el estándar IEEE 802.15.4 para las capas PHY y MAC, y por ser necesaria la se hace necesaria la implementación de un protocolo que permita complementar nuestro esquema basado el modelo OSI con una capa de red que permita procesos de enrutamiento dentro de la red. Con esta finalidad se hizo un análisis de los protocolos MIWI, MIWI P2P y ZigBee los cuales se describe a continuación:

#### 2.2.3.1. MIWI

Es un protocolo sencillo, creado por Microchip Technology Incorporated, diseñado para redes inalámbricas de baja transferencia de datos, corto alcance y bajo costo. Fundamentalmente basado en el estándar IEEE 802.15.4 para WPANs (Wireless PANs). MIWI esta diseñado para aplicaciones que impliquen redes de pequeño tamaño, que no requieran un número muy grande de dispositivos, pocos o ningún hop (puente intermedio) para lograr comunicaciones dispositivo - coordinador.

Algunas características de este protocolo son: un máximo de 1024 nodos, cada coordinador puede tener un máximo de 127 nodos asociados a él, un máximo de 8 coordinadores, 4 hops dentro de la red y 2 hops desde el coordinador.

El protocolo MIWI basa sus capas inferiores en el estándar IEEE 802.15.4 y provee de la Capa de Red para procesos de enrutamiento.

### 2.2.3.2. MIWI P2P

Este protocolo al Igual que MIWI, fue creado por Microchip Technology Incorporated, además de estar orientado al mismo tipo de redes de baja transferencia de datos, corto alcance y bajo costo. Al igual que MIWI está basado en el estándar IEEE 802.15.4 para la capa PHY y la MAC.

A diferencia de MIWI, el cual provee de una capa de red adicional, este protocolo modifica la capa MAC para hacer de esta una capa híbrida entre la capa MAC y la de Red, facilitando los procesos de reconocimiento, conexión y desconexión, entre otros, mediante la adición de comandos a la capa MAC.

Este protocolo permite  $2^{64}$  nodos, sin embargo, no permite procesos multi-hop por lo que la red se ve limitada al alcance de los dispositivos.

### 2.2.4. Protocolo ZigBee

En 2004 ZigBee Alliance, una comunidad internacional sin fines de lucro, estableció el protocolo Zigbee, el cual esta diseñado para utilizar las características del estándar 802.15.4, en especial la aplicación a redes inalámbricas de controles y sensores, de bajo costo y bajo consumo con baja tasa de transmisión de datos.

El protocolo ZigBee se caracteriza por un requerimiento de memoria de entre 4 y 32 KB, una vida útil en dispositivos a batería de años, mas de 65 mil nodos conectados por cada dispositivo maestro, una tasa de transferencia de aproximadamente 250Kb/s y un alcance de 300m, así mismo, ZigBee tiene la posibilidad de manejar una gran variedad de dispositivos clasificados por sus capacidades, estos se pueden observar en el Cuadro 2.6.

El protocolo ZigBee define la capa de red (NWK), esta se encarga de permitir a una red, con transmisores de baja potencia, crecer en tamaño a través de los procesos de enrutamiento, sufriendo de bajas latencias.

**Estructura de la NWK** La NWK consta de dos componentes, estas son: el encabezado de la capa de red (NWK Header) y la carga (NWK Payload) mostrados en la Figura 2.22, así como la composición de cada una de estas partes.

Octets: 2	2	2	1	1	Variable
Frame Control	Destination Address	Source Address	Radius	Sequence Number	Frame Payload
	Routing Fields				
NWK Header					NWK Payload

Figura 2.22: Formato general de la NWK  
Referencia: [15]

Cuadro 2.6: Dispositivos Zigbee  
Referencia: [15]

Tipo de Dispositivo	Descripción
FFD	(Full Function Device). Contiene todas las características del estándar 802.15.4, incluyendo la seguridad.
FFDNGTS	Igual que FFD pero sin capacidad para GTS.
FFDNB	Igual que FFD pero sin capacidad para Beacons.
FFDNBNS	Igual que FFD pero sin capacidad para Beacons ni Seguridad.
RFD	(Reduced Function Device). Contiene todas las características RFD del estándar 802.15.4.
RFDNB	Igual que RFD pero sin capacidad para Beacons.
RFDNBNS	Igual que RFD pero sin capacidad para Beacons ni Seguridad.

A continuación se procede a realizar una descripción de los campos que la componen.

- Control de Trama (Frame Control): Es un campo de dos octetos de longitud y contiene información que indica el tipo de trama de la comunicación, estos pueden ser observados en la Cuadro 2.7, el cual debe mantenerse en 1 para especificar que la versión usada es la 1.0.0 del protocolo, posteriormente el tipo de operación del proceso de enrutamiento, que puede ser una supresión de descubrimiento de ruta, habilitación de descubrimiento de ruta o forzar un descubrimiento de ruta, por ultimo la indicación de la presencia o no de los procesos de seguridad. Esta información se puede observar en la Figura 2.23.

Cuadro 2.7: Valores del campo de tipo de trama  
Referencia: [15]

Valor del tipo de trama b1b2	Tipo de trama
00	Data
01	Comando de red
10-11	Reservado

- Dirección de destino (Destination Address): Consta de 2 octetos que representan la dirección corta del nodo de destino final de la ruta.
- Dirección de origen (Source Address): Consta de 2 octetos que representan la *short address* del nodo que origino el envío del paquete.

Bits: 0-1	2-5	6-7	8	9	10-15
Frame type	Protocol version	Discover route	Reserved	Security	Reserved

Figura 2.23: Control de trama de la capa de red  
Referencia: [15]

- Radius: Consta de 1 octeto que indica el número de re-transmisiones que podrá hacer el paquete, este valor deberá ser decrementado en 1 tras cada retransmisión.
- Carga de la trama (Frame Payload): Posee una longitud variable según sea el tipo de trama que sea transmitida.

#### 2.2.4.1. Formatos de tramas

**1. Trama de datos** Debe seguir la estructura mostrada en la Figura 2.24, donde el control de tramas y los parámetros de ruta deben contener valores apropiados.

Octets: 2	6	Variable
Frame control	Routing fields	Data payload
NWK header		NWK payload

Figura 2.24: Trama de datos de la capa de red  
Referencia: [15]

- Encabezado de la capa de red (NWK Header): En el campo del *frame control*, el sub-campo que indica el tipo de trama debe contener el valor que indica que esta es una trama de datos, como se indica en la Cuadro 2.7. Los demás campos deberán establecerse de acuerdo al uso que se le vaya a dar a la trama. Los campos de enrutamiento deberán contener la combinación apropiada en los campos de direcciones y broadcast, dependiendo de la configuración en el *frame control*.
- Carga de la capa de red (NWK Payload): Debe contener la secuencia de octetos que las capas superiores hayan solicitado, a la capa de red, transmitir.

**2. Trama de Comando** Debe seguir la estructura mostrada en la Figura 2.25, donde el control de trama y los parámetros de ruta deben contener valores apropiados y el *payload* contendrá primeramente, en un octeto, el comando a transmitir seguido por la información asociada a dicho comando.

<b>Octets: 2</b>	<b>6</b>	<b>1</b>	<b>Variable</b>
Frame control	Routing fields	NWK command identifier	NWK command payload
NWK header		NWK payload	

Figura 2.25: Trama de datos de la capa de red  
Referencia: [15]

- Encabezado de la capa de red (NWK Header). En el campo del *frame control*, el subcampo que indica el tipo de trama debe contener el valor que indica que esta es una trama de comandos, como se indica en la Cuadro 2.7. Los demás campos deberán establecerse de acuerdo al uso que se le vaya a dar a la trama. Los campos de enrutamiento deberán contener la combinación apropiada en los campos de direcciones y broadcast, dependiendo de la configuración en el *frame control*.
- Identificador del comando de red (NWK Command Identifier). Indica el tipo de comando usado. Este campo debe contener uno de los comandos no reservados enlistados en la Cuadro 2.8
- Carga de la capa de red (NWK Payload). Debe contener el comando propiamente dicho.

La lista de comandos se muestra en la Figura 2.8

Cuadro 2.8: Tipos de comandos de la capa de red  
Referencia: [15]

Identificador	Comando
0x01	Solicitud de ruta
0x02	Respuesta de ruta
0x03	Error de ruta
0x04	Abandono
0x00, 0x05-0xff	Reservado

#### 2.2.4.2. Tramas de comandos

**1. Comando de solicitud de rutas** El comando de solicitud de rutas permite a un dispositivo solicitar que otros en el rango de alcance, inicien el proceso de búsqueda de un destino en particular y establecer una ruta de comunicacional al mismo de la manera energéticamente más económica y fácil para comunicaciones futuras. El *payload* debe tener la estructura mostrada en la Figura 2.26.

<b>Octets: 1</b>	<b>1</b>	<b>1</b>	<b>2</b>	<b>1</b>
Command frame identifier	Command options	Route request identifier	Destination address	Path cost
NWK payload				

Figura 2.26: Solicitud de rutas  
Referencia: [15]

- Opciones de comando (Command Options): El formato del octeto se muestra en la Cuadro . Este debe ser fijado el 0x01 solo si la solicitud de ruta se ha generado como parte de un proceso de reparar una ruta.

Cuadro 2.9: Campo de opciones de comandos  
Referencia: [15]

<b>Bit: 0-6</b>	<b>7</b>
<b>Reservado</b>	<b>Reparacion de ruta</b>

- Identificador de solicitud de ruta (Route Request Identifier): Es un número secuencial de 8 bits que indica la cantidad de solicitudes que se han realizados, este es incrementado en uno cada vez que la capa de red realiza este proceso.
- Dirección de destino (Destination Address): Son dos octetos que deben contener el identificador del destino previsto para este proceso.
- Path Cost: Es un octeto que se usa para acumular el costo, en términos de eficiencia de transmisión de dato, que acarrea la solicitud mientras se mueve por la red.

**2. Comando de aceptación de ruta** Permite al dispositivo de destino, que ha sido especificado en el comando de solicitud de ruta previamente realizado, informar al generador de la solicitud de la ruta que esta ha sido recibida. Así mismo permite a los enrutadores establecer el enlace de comunicación entre ambos dispositivos usando el camino energéticamente más eficiente. El *payload* debe tener el formato establecido en la Figura 2.27.

<b>Octets: 1</b>	<b>1</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>1</b>
Command frame identifier	Command options	Route request identifier	Originator address	Responder address	Path cost
NWK payload					

Figura 2.27: Aceptación de ruta  
Referencia: [15]

- Command options: Este tiene las mismas especificaciones mencionadas en el comando de solicitud de rutas.
- Route request identifier: Debe corresponder con el identificador de la solicitud de rutas a la cual esta respondiendo.
- Dirección del generador (Originator Address): Es un campo de dos octetos de longitud que debe contener el *short address* del dispositivo para el cual se esta estableciendo la ruta.
- Path cost: Debe contener el valor total del costo, en términos de eficiencia de transmisión de dato, que acarrea la comunicación con el dispositivo a través de la ruta seleccionada.

**3. Comando de error de rutas** Este error sucede cuando un dispositivo falla en el proceso de retransmisión de un dato, el comando informa al origen del paquete que la ruta a fallado. El *payload* debe tener el formato establecido en la Figura 2.28.

Octets: 1	1	2
Command frame identifier	Error code	Destination address

Figura 2.28: Error en ruta  
Referencia: [15]

- Código de error (Error Code): Debe contener uno de los valores no reservados mostrados en la Cuadro 2.10.
- Dirección de destino: Debe contener el *short address* de destino de la trama de datos que encontró el error.

Cuadro 2.10: Códigos de error  
Referencia: [15]

Identificador	Comando
0x00	No route available
0x01	Tree link failure
0x02	Non-Tree link failure
0x03	Low battery level
0x04	No route capacity
0x00, 0x02-0xff	Reserved

**4. Comando de abandono de rutas** Este comando informa al padre del dispositivo que este va a salir de la red o para solicitar su salida. El *payload* debe tener el formato establecido en la Figura 2.29.

Octets: 1	1
Command frame identifier	Command options

Figura 2.29: Abandono de ruta  
Referencia: [15]

### 2.2.4.3. Protocolo de enrutamiento usado en ZigBee

Uno de los protocolos usado en ZigBee es el denominado Ad-hoc On-Demand Distance Vector (AODV, Vector de distancia para saltos bajo demanda), como su nombre lo indica este algoritmo solo genera rutas de comunicación hasta que estas no sean necesarias, este tipo de algoritmo es el usado en las redes de tipo Mesh.

El proceso inicia cuando un dispositivo debe comunicarse con otro que no se encuentra en su rango de alcance. En esta situación se inicia el proceso de solicitud de rutas, todos los nodos al alcance reciben el paquete y almacenan la información del dispositivo del cual recibieron esta solicitud en caso de ser escogidos como parte de la ruta y retransmiten el paquete hasta el nodo de destino solo si la ruta es más eficiente que alguna que ya hayan recibido, esto evita que ocurran redundancias que puedan resultar en un incremento de la latencia, gasto innecesario de energía y ocupación del canal indiscriminado.

Una vez la solicitud haya alcanzado al nodo de destino este responde usando el camino mas eficiente hasta llegar al dispositivo para el cual se estableció la ruta.

En caso de que la ruta llegase a fallar en algún punto, se tendría que repetir todo el proceso.

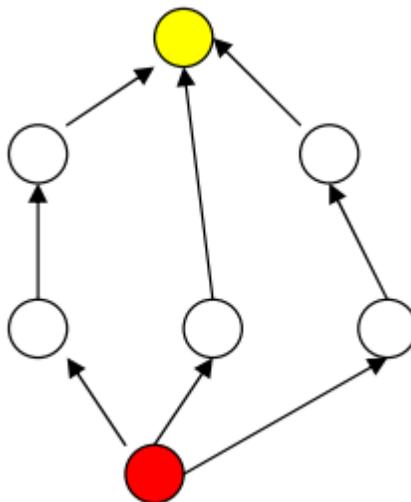


Figure 2.30: Proceso de solicitud de ruta

## 2.2.5. Módulos Compatibles con el Estándar

### 2.2.5.1. ZigBit ATZB-24-A2

Este módulo, mostrado en la Figura??, fue desarrollado por la Casa Atmel, diseñado bajo el estándar IEEE 802.15.4.

Fue diseñado con el fin de cumplir tareas básicas, que requieran el mínimo de energía. Está constituido por un microcontrolador ATmega1281 y un módulo transreceptor AT86RF230 con la finalidad de ofrecer una plataforma lista para implementación y disminuir el tiempo de salida al mercado de proyectos cuyos fabricantes incluyan este módulo en su diseño.

Para tener una idea de la constitución de este módulo se puede observar, en la Figura 2.31, el diagrama de bloques ofrecido por el fabricante.

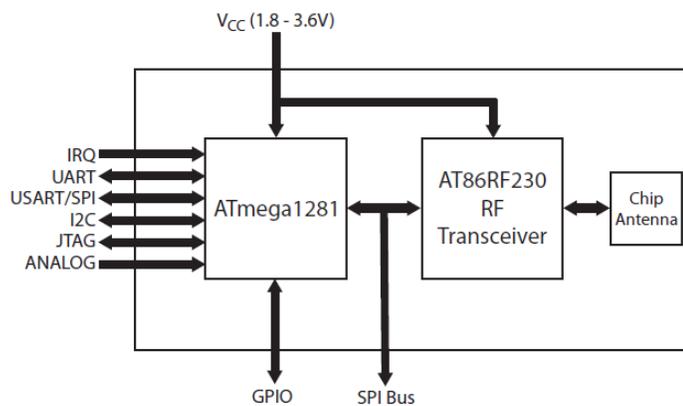


Figura 2.31: Diagrama de de Bloques del módulo ZigBit ATZB-24-A2.

Entre algunas de sus características de interés se tienen:

- Dimensiones en milímetros de: 24x13.5x2.0
- Tensión de alimentación de 1.8 a 3.6 V.
- Consumo de corriente en emisión de 18mA.
- Consumo de corriente en recepción de 19mA.
- Consumo de corriente en modo ahorro de energía de  $6\mu\text{A}$ .
- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de -17dBm a 3dBm en 16 niveles.
- Sensibilidad de -101dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: UART, USART, SPI, I<sup>2</sup>C y 1-Wire.

- Soporta ZigBee.
- Capacidad de manejo de topología tipo Mesh.
- Costo actual por unidad de 54.54 dólares (USA).

En la Figura 2.32 se muestra el patrón de radiación del ZigBit ATZB-24-A2 en su plano acimutal y de elevación.

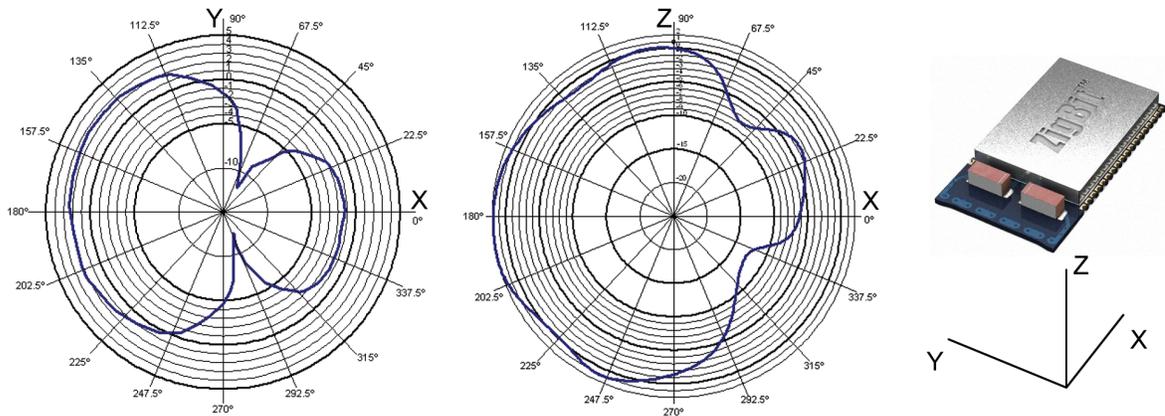


Figura 2.32: Patrón de radiación del módulo ZigBit ATZB-24-A2. A la izquierda se muestra el plano acimutal, en el centro el plano de elevación y a la derecha la referencia con respecto al módulo.

Referencia: [16]

### 2.2.5.2. XBee



Figura 2.33: Módulo de comunicación inalámbrica XBee.

Este módulo de comunicaciones en RF, presentado en la Figura 2.33, fue desarrollado por Digi International, diseñado bajo el estándar IEEE 802.15.4 y orientado a redes de baja transferencia de datos. Como muestra su diagrama de bloques, el cual se puede observar en la Figura 2.34, mediante el uso de su UART permite la transferencia de información con dispositivos externos, información que luego es llevada a los módulos de transmisión RF.

Físicamente está compuesto por un módulo de comunicación inalámbrica fabricado por Freescale, modelo MC13193, responsable de la estructuración de la capa PHY y la MAC, además posee un microcontrolador, el M9S08GT60, fabricado por la misma casa, para el control del resto de sus periféricos. Al igual que el ZigBit ATZB-24-A2 este módulo fue diseñado para ser una herramienta lista que permita muestrear variables, procesar información, entre otros.

Dentro de alguna de las características de interés de este módulo se pueden mencionar:

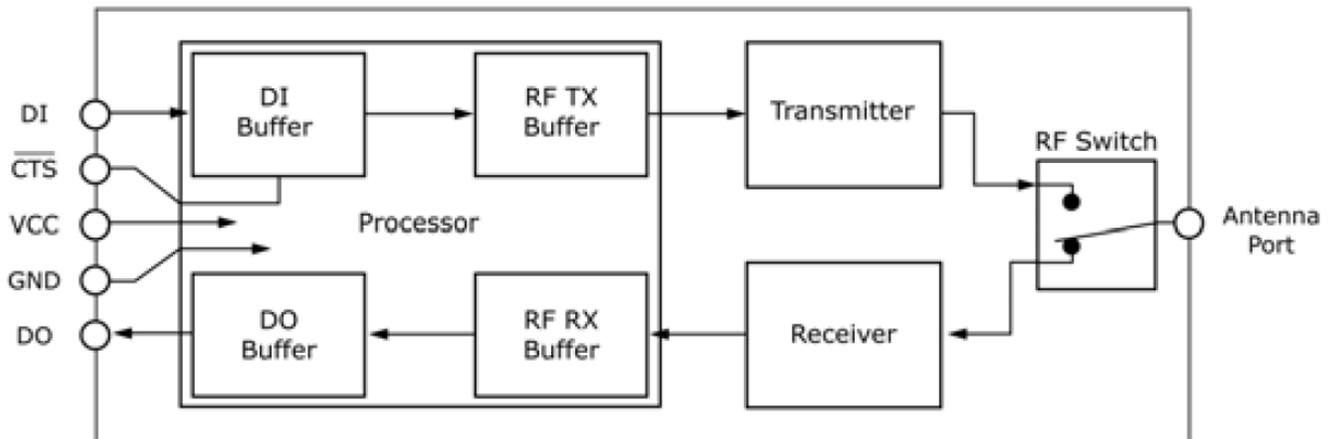


Figura 2.34: Diagrama de bloques del módulo XBee.

- Dimensiones en milímetros: 24.38x27.61x2.79 (no tomando en cuenta la longitud de los pines)
- Tensión de alimentación de 2.8 a 3.4 V.
- Consumo de corriente en emisión de 45mA.
- Consumo de corriente en recepción de 50mA.
- Consumo de corriente en modo ahorro de energía de  $10\mu A$ .
- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de -10dBm a 0dBm en 5 niveles.
- Sensibilidad de -92dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: UART.
- Soporta ZigBee.
- Capacidad de manejo de topologías tipo punto-a-punto, punto-multipunto, peer-to-peer y mesh.

- Costo actual por unidad de 28.95 dólares (USA).

En la Figura 2.35 se muestra el plano acimutal del patrón de radiación del módulo de comunicaciones XBee.

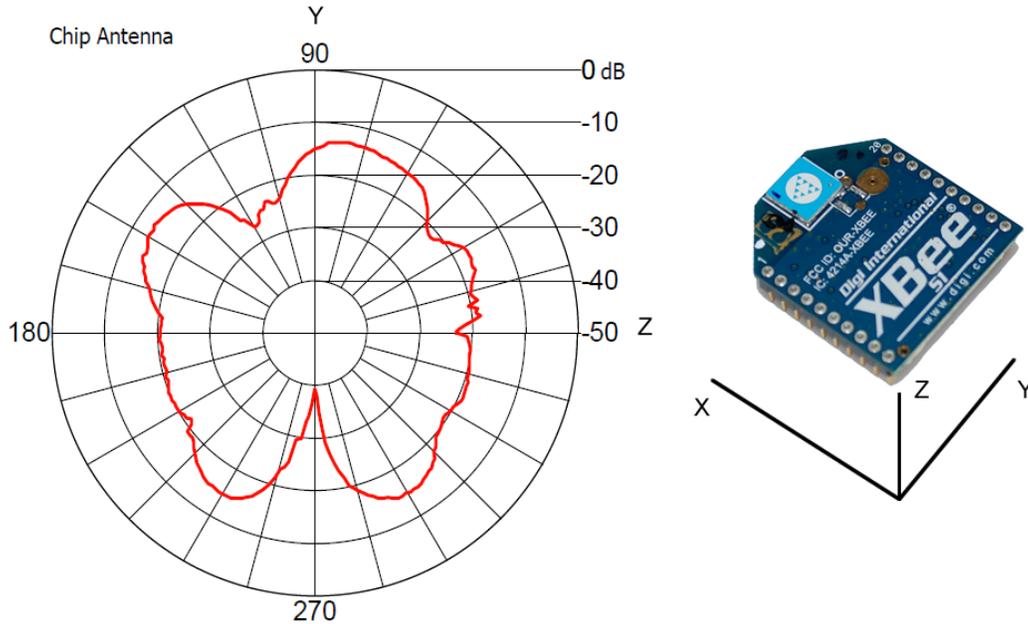


Figura 2.35: Plano acimutal del patrón de radiación de la antena tipo chip del módulo XBee.

Referencia: [17]

### 2.2.5.3. XBeePro



Figura 2.36: Módulo de comunicación inalámbrica XBee PRO.

El módulo XBeePro, mostrado en la Figura al igual que el XBee, fue diseñado por Digi International, bajo los mismos criterios salvo que con este modelo se busca un mayor alcance. Se listan a continuación algunas de sus características de interés para este trabajo de grado:

- Dimensiones en milímetros: 24.38x32.94x2.79 (no tomando en cuenta la longitud de los pines)
- Tensión de alimentación de 2.8 a 3.4 V.
- Consumo de corriente en emisión de 150mA.
- Consumo de corriente en recepción de 180mA.
- Consumo de corriente en modo ahorro de energía de  $10\mu\text{A}$ .
- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de -3dBm a 10dBm en 5 niveles.
- Sensibilidad de -100dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: UART.
- Soporta ZigBee.
- Capacidad de manejo de topologías tipo punto-a-punto, punto-multipunto, peer-to-peer.
- Costo actual por unidad de 56.95 dólares (USA).

Cabe destacar que el XBeePro posee el mismo diagrama de bloques y el mismo patrón de radiación del XBee.

#### 2.2.5.4. ProFLEX01



Figura 2.37: Módulo de comunicación inalámbrica ProFlex01.

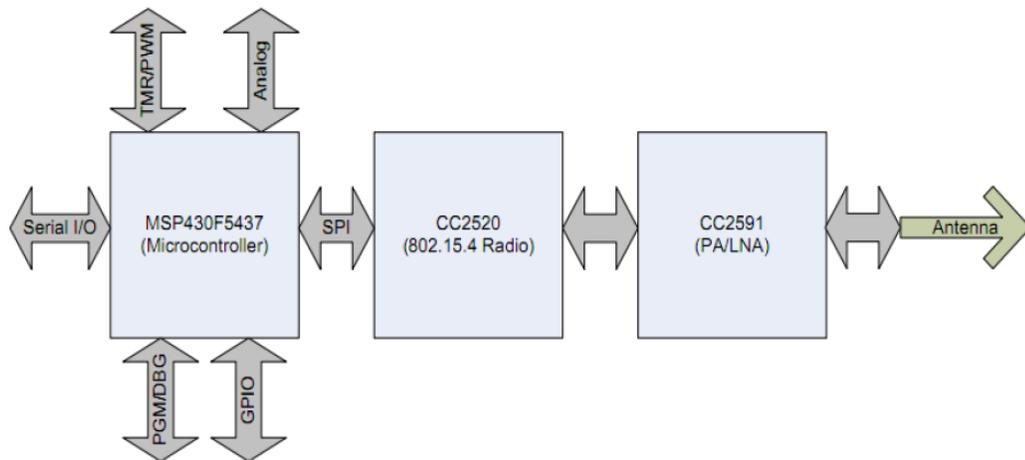


Figura 2.38: Diagrama de bloques del módulo de comunicaciones ProFLEX01.

Es un módulo creado por LS Research, está constituido por un módulo de comunicaciones CC3530 y un microcontrolador MSP430F5437 ambos de la Texas Instruments, los cuales se muestra dentro del diagrama de bloques visto en la Figura 2.38.

Entre algunas de sus características de interés resaltan:

- Dimensiones en milímetros: 22.86x41.40x5.08
- Tensión de alimentación de 2.2 a 3.3 V.
- Consumo de corriente en emisión de 145mA.
- Consumo de corriente en recepción de 30mA.
- Consumo de corriente en modo ahorro de energía de  $8\mu A$ .
- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de 7dBm a 20dBm en 5 niveles.
- Sensibilidad de -98dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: SPI.
- Soporta ZigBee.
- Capacidad de manejo de topologías tipo punto-a-punto, punto-multipunto, peer-to-peer.
- Costo actual por unidad de 22.49 dólares (USA).

El fabricante no incluye diagramas del patrón de radiación de este módulo dentro de su hoja de especificaciones.

## 2.2.5.5. MRF24J40MB



Figura 2.39: Módulo de comunicación inalámbrica MRF24J40MB.

Este módulo fue fabricado por la Microchip Technology Incorporated, fue diseñado bajo las especificaciones del estándar IEEE 802.15.4. Cuenta con un dispositivo interno encargado del control de los procesos de radio frecuencia, el MRF24J40 fabricado por la misma empresa. Sin embargo, a diferencia de los módulos mencionados anteriormente, este no cuenta con un microcontrolador que procese la información recibida de forma inalámbrica y de igual forma le transmita la data a ser enviada vía RF, por lo tanto, necesita de un microcontrolador que le brinde dichas características.

Dentro de algunas de las características de interés se pueden mencionar:

- Dimensiones en milímetros: 22.86x33.02x3.35
- Tensión de alimentación de 2.4 a 3.6 V.
- Consumo de corriente en emisión de 130mA.
- Consumo de corriente en recepción de 25mA.
- Consumo de corriente en modo ahorro de energía de  $5\mu A$ .
- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de -36dBm a 20dBm en 32 niveles.
- Sensibilidad de -102dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: SPI.
- Soporta MIWI, MIWI P2P, ZigBee y Protocolos creados por el usuario.
- Capacidad de manejo de topologías tipo Estrella, Peer-to-Peer y Mesh.
- Costo actual por unidad de 17.82 dólares (USA).

En la Figura 2.40 se muestra el patrón de radiación de la antena F del módulo MRF24J40MB.

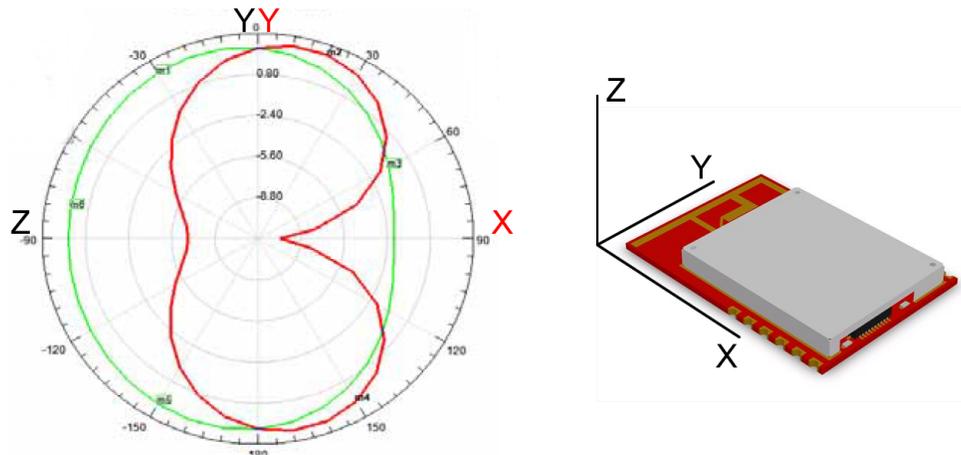


Figura 2.40: Planos acimutal (Rojo) y de elevación (Verde) del Patrón de Radiación del módulo MRF24J40MB.

Referencia: [18]

#### 2.2.5.6. MRF24J40MA

Este módulo fue fabricado por la Microchip Technology Incorporated, fue diseñado bajo las especificaciones del estándar IEEE 802.15.4. Se podría decir que es la versión de baja potencia del MRF24J40MA ya que fue concebido bajo los mismos parámetros salvo que los niveles de potencia de este módulo son mucho menores.



Figura 2.41: Módulo de comunicación inalámbrica MRF24J40MA.

- Dimensiones en milímetros: 17.78x27.94.40x1.70
- Tensión de alimentación de 2.4 a 3.6 V.
- Consumo de corriente en emisión de 23mA.
- Consumo de corriente en recepción de 19mA.
- Consumo de corriente en modo ahorro de energía de  $2\mu\text{A}$ .

- 16 canales en la banda ISM 2.4GHz.
- Potencia ajustable de salida de -36dBm a 0dBm en 32 niveles.
- Sensibilidad de -94dBm.
- Ancho de banda de 250Kbps.
- Interfaces de comunicación: SPI.
- Soporta MIWI, MIWI P2P, ZigBee y protocolos creados por el usuario.
- Capacidad de manejo de topologías tipo estrella, peer-to-peer y mesh.
- Costo actual por unidad de 9.12 dólares (USA).

**1. Consideraciones de diseño y montaje** Este módulo de comunicaciones posee 12 terminales como se muestra en la Figura 2.42, los cuales serán descritos a continuación:

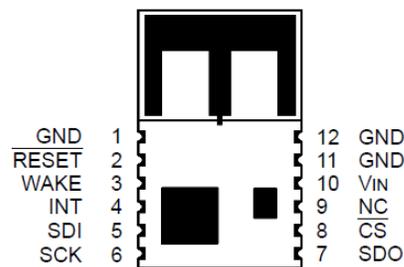


Figura 2.42: Esquema de pines del MRF24J40.

- Terminales 1, 11 y 12 (*GND*): Tierra.
- Terminal 2 ( $\overline{RESET}$ ): Reset global del módulo. Este funciona con lógica negada.
- Terminal 3 (*WAKE*): Estímulo externo para "sacar" al módulo de comunicaciones del modo de bajo consumo.
- Terminal 4 (*INT*): Indica al microcontrolador externo que un evento a ocurrido.
- Terminal 5 (*SDI*): Terminal de recepción de datos vía comunicación SPI.
- Terminal 6 (*SCK*): Señal de reloj requerida para la comunicación SPI.
- Terminal 7 (*SDO*): Terminal de emisión de datos vía comunicación SPI.
- Terminal 8 ( $\overline{CS}$ ): Habilitación de la comunicación SPI. Este funciona con lógica negada
- Terminal 9 (*NC*): Terminal de no conexión. NO DEBE CONECTARSE.
- Terminal 10 (*Vin*): Alimentación de entrada.

El fabricante sugiere para mejor desempeño la colocación de un plano de tierra del lado contrario de la placa de circuito impreso de donde será soldado el módulo. Sugiere al menos 0.4 pulgadas de cada lado del módulo además de mantener la sección de la antena fuera de la tarjeta. De igual forma indica una región de zona cercana que debe mantenerse libre de objetos metálicos. Todos estos aspectos son señalados gráficamente en la Figura 2.43.

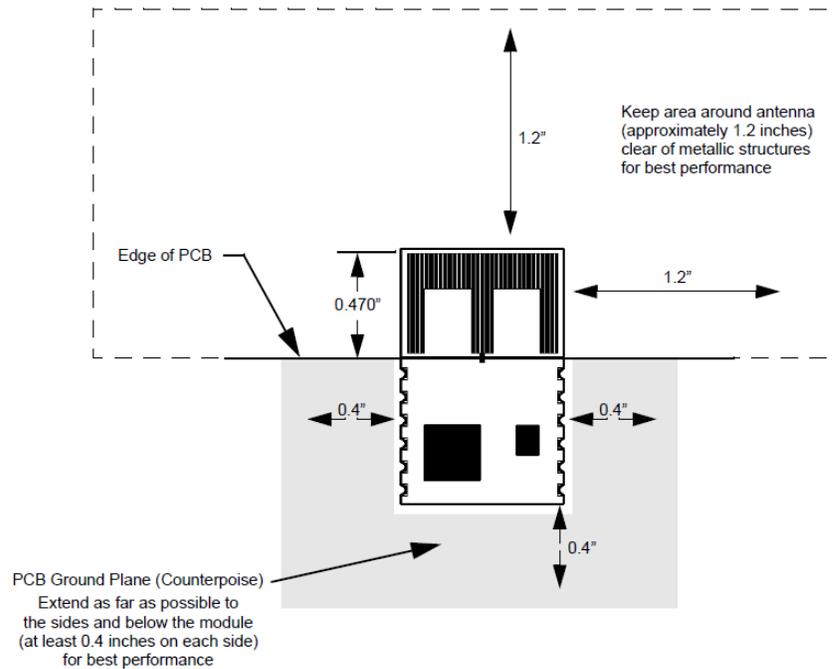


Figura 2.43: Diagrama de diseño y montaje del módulo de comunicaciones *MRF24J40MA*.

**2. El patrón de radiación** El patrón de radiación mostrado por el fabricante, puede observarse en la Figura 2.44. Se observa su en plano acimutal, de color negro, que la antena describe una emisión omnidireccional y en el corte de elevación, en rojo, que se forman dos lóbulos en dirección frontal y posterior del módulo de comunicaciones.

**3. Interfaz SPI** El microcontrolador MRF24J40 es capaz de realizar una comunicación comunicación SPI que funciona de interfaz entre el módulo de comunicaciones MRF24J40MA y el microcontrolador externo.

Primeramente, hay que señalar que el MRF24J40 posee dos secciones en memoria, la primera esta dedicada a los registros de control y posee un direccionamiento corto, la otra, por el contrario, posee un direccionamiento largo.

**Lectura y escritura de registros de direccionamiento corto.** Inicialmente es necesario colocar el terminal *chip select* en cero para habilitar la interfaz SPI, luego se arma el paquete a enviar escribiendo un cero(0) en el bit mas significativo del primer octeto a enviar, indicando que es una operación destinada a un registro de direccionamiento corto, seguido de seis(6) bits correspondientes a la dirección del registro y finalmente un octavo bit que indica



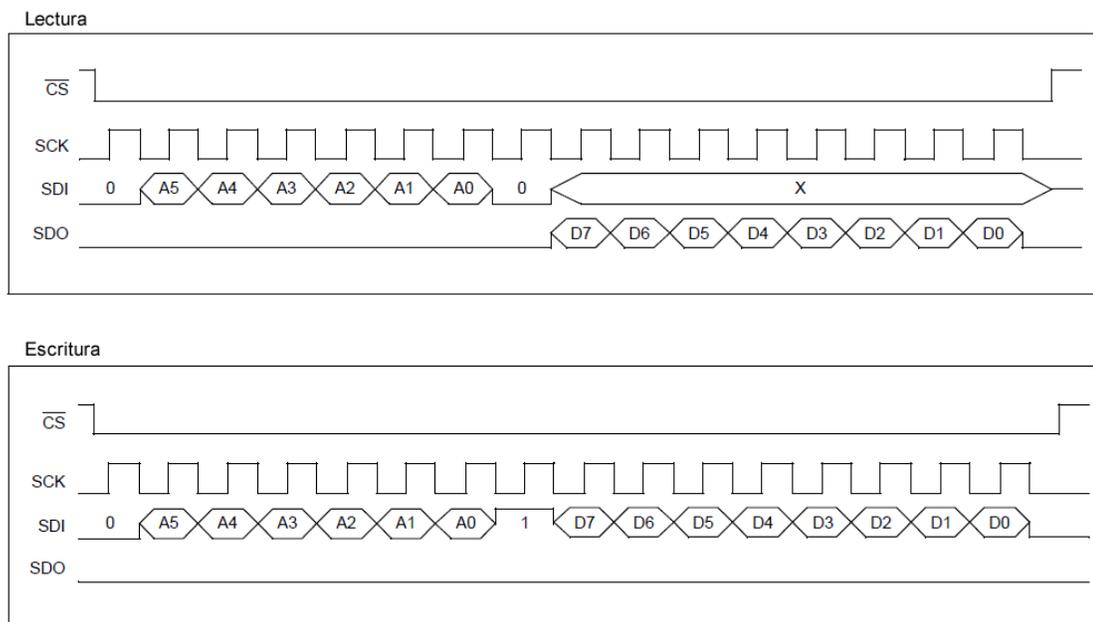


Figura 2.45: Lectura y escritura de registros de corto direccionamiento del MRF24J40.

muestran en la Figura 2.45.

**4. Configuración** Para poder trabajar con el MRF24J40MA es necesario pasar por un proceso de inicialización donde se configuran una lista de parámetros dependiendo del tipo de dispositivo ya sea un coordinador de PAN, coordinador, o un dispositivo.

Primeramente se induce un *reset* por software para lograr un estado estable en el MRF24J40 escribiendo en el registro SOFTRST (0x0D) el valor 0x07. Seguidamente se configuran valores iniciales para el identificador de PAN (2 bytes), identificador corto (2 bytes), e identificador largo (8 bytes). Estos se escriben en los registros PANIDL y PANIDH, SADRL y SADRH, y del EADR0 al EADR7 respectivamente. Cabe destacar que el identificador largo o extendido también es conocido como *MAC address* y son regulados por la IEEE [14]. Cada dispositivo en el mundo posee identificador extendido único.

Finalmente, se configuran los demás parámetros como se muestra en el Cuadro 2.11. Queda de parte del lector, el detalle de dicha configuración dentro de las especificaciones del MRF24J40MA.

**5. Recepción** El MRF24J40 maneja independientemente la componente SHR de la capa física por lo que el acceso a los procesos de sincronización no están permitidos, sin embargo, si permite el acceso al resto de las componentes de la capa física y las capas contenidas en ella. Se puede observar en la Figura 2.47, la estructura básica de un paquete recibido donde se marca la componente SHR de la capa PHY, que es removida previamente antes de permitir el acceso al paquete recibido, y de igual forma se marcan los campos que son acuñados en la cola del paquete en el momento de su recepción. Finalmente se muestra la composición del paquete recibido, contenido en el *buffer* de recepción RXFIFO.

Se describe a continuación el proceso de recepción de un paquete:

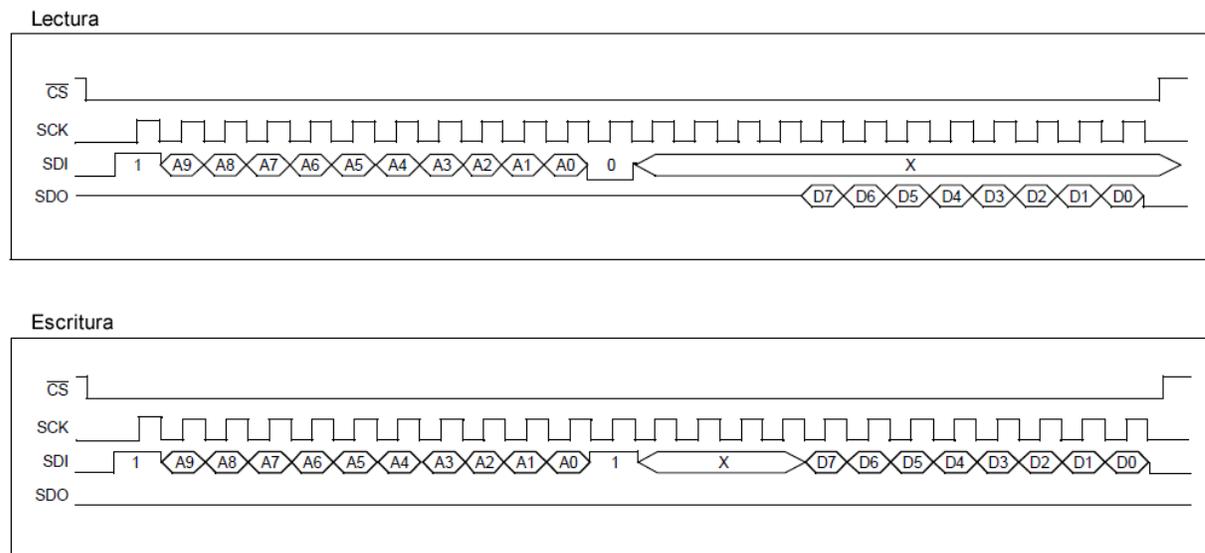


Figura 2.46: Lectura y escritura de registros de corto direccionamiento del MRF24J40.

- Se recibe una interrupción por parte del MRF24J40 poniendo el pin INT en alto.
- Inhabilitar la recepción de paquetes en aire por parte del MRF24J40MA. Esto se logra poniendo en alto el bit RXDECINV del registro BBREG1 (0x39).
- Leer el registro INTSTAT (0x31) para determinar el origen de la interrupción del MRF24J40MA.
- Verificar que la bandera RXIF del INTSTAT está en alto, la cual indica que ha sido recibido un paquete.
- Leer la posición en memoria 0x0300 para obtener la longitud del paquete recibido. El *buffer* de recepción se denomina RXFIFO y posee una capacidad de 128 bytes. La longitud del paquete recibido contenido en el RXFIFO se denomina *frame length*.
- Leer las posiciones desde la 0x0301 hasta la  $0x0301 + \textit{frame length} + 2$ . En estos 2 bytes adicionales se leen el LQI y el RSSI<sup>1</sup> respectivamente.
- Habilitar nuevamente la recepción de paquetes en aire, poniendo en cero(0) el bit RXDECINV del registro BBREG1 (0x39).

**6. Transmisión** La transmisión de paquetes es un poco más compleja que la recepción, por la variedad de tramas que se manejan dentro del estándar. Debido a esto, se han asignado secciones de la memoria del MRF24J40 para los diferentes tipos de trama como se muestra en la Figura 2.48. Cabe destacar que no se le fue asignado ningún espacio de memoria a los

<sup>1</sup>El RSSI es un parámetro que indica el nivel de potencia de la señal recibida. Puede tomar valores de 0 a 255.

Cuadro 2.11: Configuración de los parámetros de control del MRF24J40MA para un dispositivo Coordinador de PAN.

Registro	Dirección	Valor	Registro	Dirección	Valor
PACON2	0x18	0x98	ORDER	0x10	0xFF
TXSTBL	0x2E	0x95	BBREG2	0x3A	0xB8
RFCON0	0x2000	0x03	CCAEDTH	0x3F	0x60
RFCON1	0x2001	0x02	BBREG6	0x3E	0x40
RFCON2	0x2002	0x80	TXBCON1	0x25	0x30
RFCON3	0x2003	0x00	TXPEND	0x21	0xFC
RFCON6	0x2006	0x90	TXTIME	0x27	0x38
RFCON7	0x2007	0x80	TXNCON	0x1B	0x00
RFCON8	0x2008	0x10	INTCON	0x32	0xF6
SLPCON0	0x0211	0x03	RFCON0	0x0200	0xF3
SLPCON1	0x0220	0x21	RFCTL	0x36	0x04
RXMCR	0x00	0x29	RFCTL	0x36	0x00
TXMCR	0x11	0x5C	RXFLUSH	0x0D	0x01

procesos de reconocimiento debido a que estos son manejados directamente por el MRF24J40 en forma automática. Sólo debe activarse esta característica en la configuración.

Para el armado del paquete se toma en cuenta el hecho de la independencia de la componente SHR, perteneciente a la PHY, por la misma razón descrita en la recepción. Por lo que el objetivo se limita a armar la componente PHR, que contiene el *frame length* del paquete a transmitir, y la PSDU que contiene el resto de las capas superiores incluyendo la MAC.

Se muestra en la Figura 2.49 los tipos diferentes de tramas y como se van armando hasta componer el paquete total a enviar.

El proceso de transmisión se lista a continuación:

- o Es necesario definir previamente todos los parámetros que van a ser configurados para la correcta transmisión del paquete.<sup>1</sup> Esto implica que tipo de identificadores serán contenidos en él, es decir: si este requiere verificación de correcta recepción mediante el uso procesos de reconocimiento, si el paquete va a ser encriptado, los tipos de identificadores que estarán contenidos en él, entre otros. Cabe señalar que se ha de prestar suma atención a cada subcampo del *frame control* del encabezado de la capa MAC. De igual forma se debe analizar qué carga va a llevar el paquete para determinar finalmente las longitudes tanto del MHR como del paquete completo. Es importante, a la hora de escribir, hacerlo correctamente según lo establecido.

<sup>1</sup>Si no es armado correctamente el paquete, el MRF24J40 lo calificará como no válido y abortará los procesos de envío.

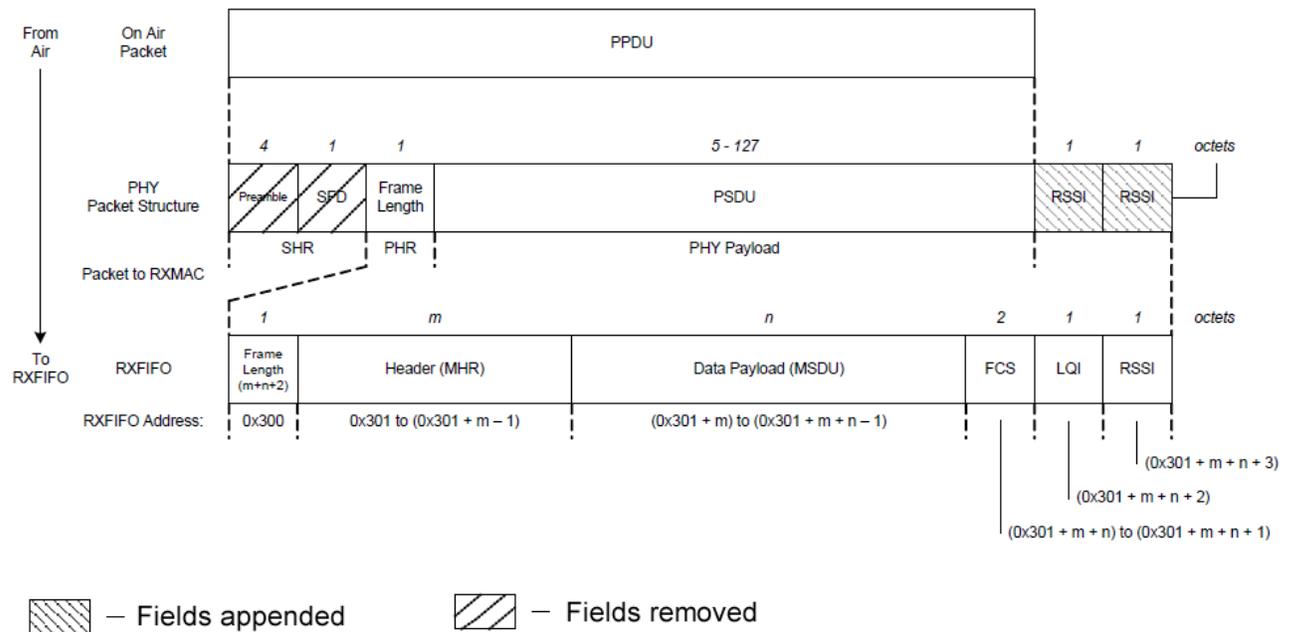


Figura 2.47: Estructura de un paquete recibido por el MRF24J40MA.

- Se arma el paquete, preferiblemente en un vector antes de ser cargado al MRF24J40, iniciando con la longitud del encabezado de la MAC en el primer byte, seguido por la longitud del paquete completo (incluyendo el Payload) en el segundo byte como se muestra en la primera trama de la Figura 2.49.
- Seguido a esto, se escribe el encabezado de la MAC según lo especificado en el *frame control*.
- Luego se define toda la carga que llevará el paquete.
- Se copia el vector armado en la memoria de transmisión correspondiente al tipo de trama. TX Beacon FIFO (Beacons), TX Normal FIFO (Datos y comandos).
- Se indica al MRF24J40 que el paquete está cargado y se procede a realizar la transmisión del mismo. Esto al igual que el paso anterior depende del tipo de trama bajo la cual se desea transmitir. Los diferentes tipos de estímulos de inicio de transmisión correspondientes a los distintos tipos de paquetes se encuentran en los registros TXBCON0 (0x1A), TXNCON (0x1B), TXG1CON (0x1C) y TXG2CON (0x1D). Cabe destacar que aquellos paquetes que requieran hacer uso de los procesos de reconocimiento deben especificarlo en los registros de estímulos de inicio de transmisión de igual forma.

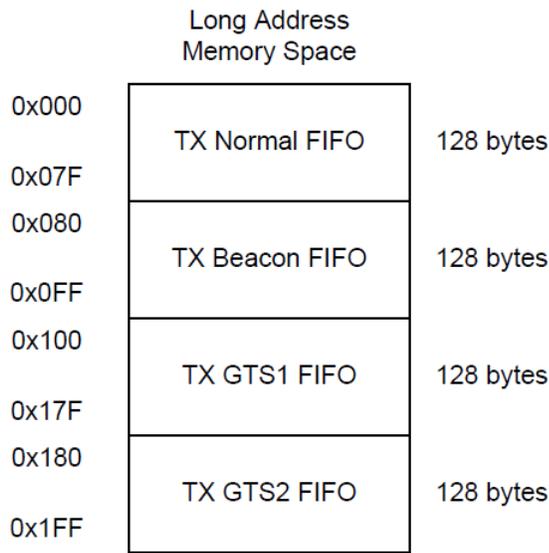


Figura 2.48: Secciones de memoria del MRF24J40 para transmisión de los diferentes tipos de trama.

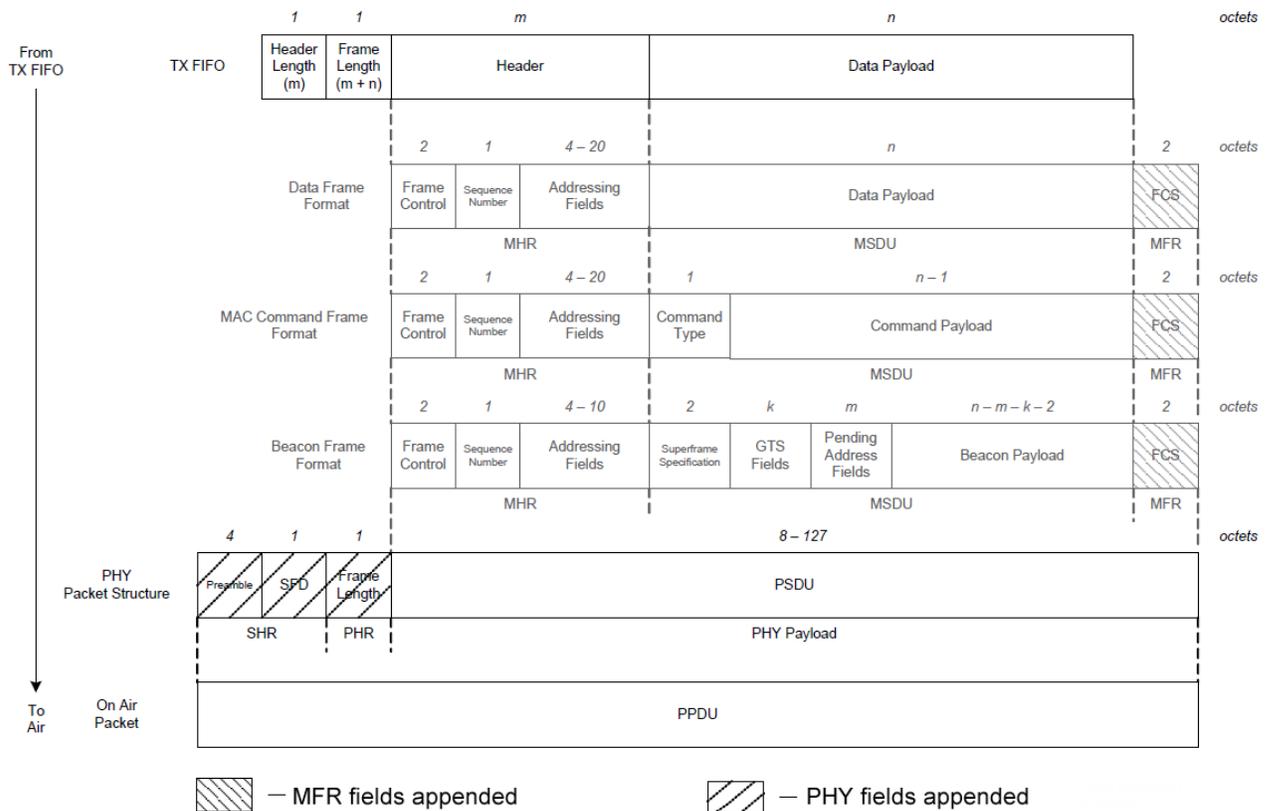


Figura 2.49: Estructura de los diferentes paquetes posibles a ser transmitidos por el MRF24J40MA.

### 2.2.6. MikroC PRO for PIC V5.6

Es un IDE desarrollado por Mikroelektronika, que siguiendo el estándar ANSI C , permite el desarrollo de programas en un lenguaje C un poco modificado para facilitar la creación de código para microcontroladores. Dispone de múltiples herramientas internas integradas para proveer de una interfaz cómoda, de fácil manejo y de gran utilidad además de disponer de un número considerable de bibliotecas de funciones que permiten el desarrollo rápido y eficiente del código.

Se puede observar en la Figura 2.50, las principales componentes de IDE lo cuales se listan a Continuación:

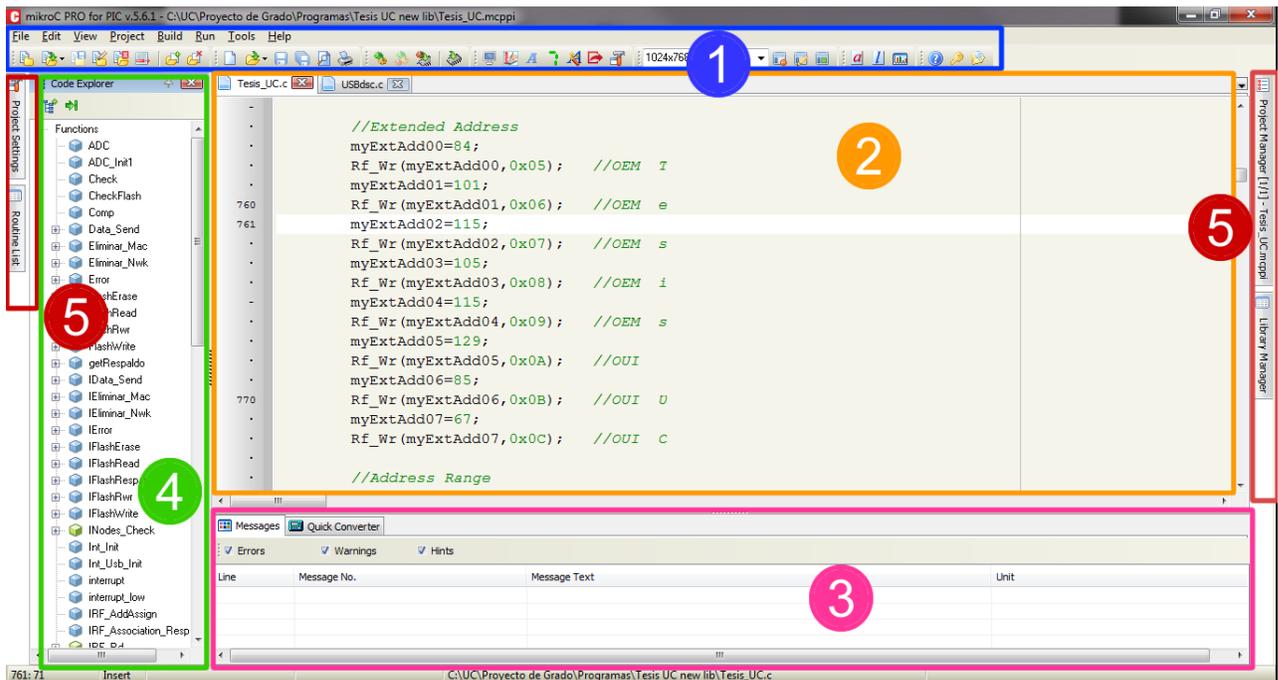


Figura 2.50: Entorno de programación del software MikroC PRO for PIC v5.6. Con el número uno se identifica la barra de herramientas y demás barras de compilación, indentación, entre otros; con el número dos el cuadro de codificación del programa, con el tres la barra de mensajes, con el cuatro el explorador de código y con el cinco las barras de acceso rápido.

- Barra de herramientas. En este sector se tiene acceso a las configuraciones de todos los tipos de opciones mediante los menús File, Edit, View, etc., entre todos ellos brindan opciones para la edición del proyecto, del código, preferencias del entorno, entre otros; a su vez contiene accesos directos a las principales funciones como compilar, guardar, cerrar proyecto y de igual forma se tiene un grupo de útiles herramientas como el generador de caracteres especiales para pantallas LCD, configuración de números para displays siete segmentos, entre otros.
- Esta es la ventana de programación, aquí es donde se escribe el código del microcontrolador en estudio. Esta ventana posee ayudas visuales para la programación como la indentación automática en función del código y variedad de colores para los dife-

rentes tipos de variables que usualmente se manejan dentro de la programación como declaraciones, comentarios, entre otros.

- c) La barra de mensajes, en la cual se puede observar todos los procesos realizados durante la compilación, duración de esta, memoria RAM libre, memoria Flash Libre entre otros. A su vez esta ventana está compartida con una herramienta llamada Quick Coverter que permite transformar un número en cualquiera de los formatos, ASCII, Hex, Dec, Bin en cualquiera de los otros.
- d) El Code Explorer es una herramienta práctica que permite el acceso rápido a todos los subprogramas desarrollados en el código, y te permite ver los parámetros de entrada de cada uno sin necesidad de llegar hasta la línea en el que este se encuentre facilitando el desarrollo del programa.
- e) Las barras de acceso rápido nos permite mantener minimizadas herramientas con usos menos frecuentes pero que siempre es bueno tener a un fácil acceso como el explorador del proyecto para agregar alguna librería, abrir un proyecto en conjunto con el actual, entre otros.

Como muchos lenguajes, el lenguaje C modificado de MikroC permite el manejo de variables enteras como lo son: bit, short, long, double, etc, todas incluyendo sus formatos tanto con como sin signo; permite el manejo de vectores, matrices, múltiples operaciones aritméticas por línea, entre otros.

Dentro de alguna de las bibliotecas de funciones de interés para este trabajo de grado se pueden mencionar: SPI en hardware y software, y USB, estas se encargan de habilitar el modulo de comunicación correspondiente y generar los procesos necesarios dentro del micro-controlador para efectuar satisfactoriamente la transmisión.

### 2.2.7. Eagle V6.3

Este es un IDE orientado al desarrollo de tarjetas de circuito impreso, cuenta con dos entornos de desarrollo, uno a partir del esquema circuital donde este se arma y luego se ensambla automáticamente la tarjeta de circuito impreso mediante algunos parámetros indicados por el usuario como: circuito más compacto posible, mejor disipación de calor, entre otros; y de igual forma se maneja un segundo entorno de desarrollo donde se trazan directamente las pistas para las diferentes capas disponibles.

En la Figura 2.51 se muestra, en entorno para el trazado de pistas utilizado para este trabajo de grado. Este cuenta con 3 componentes básicas listadas a continuación:

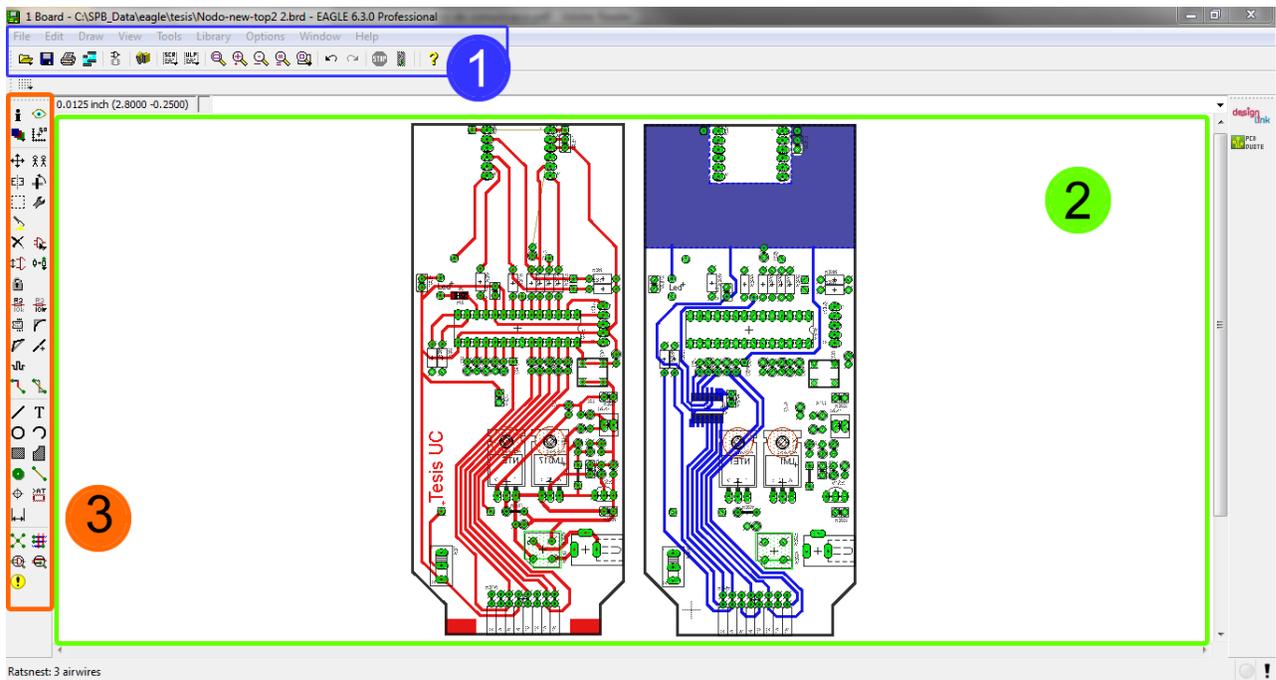


Figura 2.51: Entorno de Diseño para Tarjetas de Circuitos Impresos del Software Eagle v6.3.

- Barra de herramientas que permite la configuración de todas las herramientas y preferencias que ofrece el programa.
- En este entorno se delimitan los bordes de la tarjeta, pistas, pads, se agregan conectores de casi cualquier tipo. Entorno configurable al que se le puede cambiar el color de fondo, grilla ultra fina, entre otros.
- La paleta. Esta consta de las principales herramientas a utilizar a la hora del diseño de la tarjeta de circuito impreso, permite el cambio de capas, el acceso a las librerías de componentes entre otros.

Este entorno posee integrado un conjunto de bibliotecas para los componentes que facilitan visualmente el diseño de la tarjeta. Dentro de estas librerías se pueden encontrar esquemas de resistencias, capacitores, relays, microcontroladores de cualquier empaquetado, dispositivos de montaje superficial, entre muchos otros.

Permite exportación para archivos gerber, utilizados por la mayoría de las maquinas de CNC. Además permite la exportación de archivos en formatos .pdf para su fácil impresión en cualquier impresora láser.

La version utilizada para el desarrollo de este trabajo especial de grado fue la version de demostracion ofrecida por la empresa CadSoft.

### 2.2.8. El Microcontrolador PIC18F27J53

Es un microcontrolador fabricado por Microchip Technology Incorporated, perteneciente a la gamma 18. Posee 28 terminales como se muestra en la Figura 2.52.

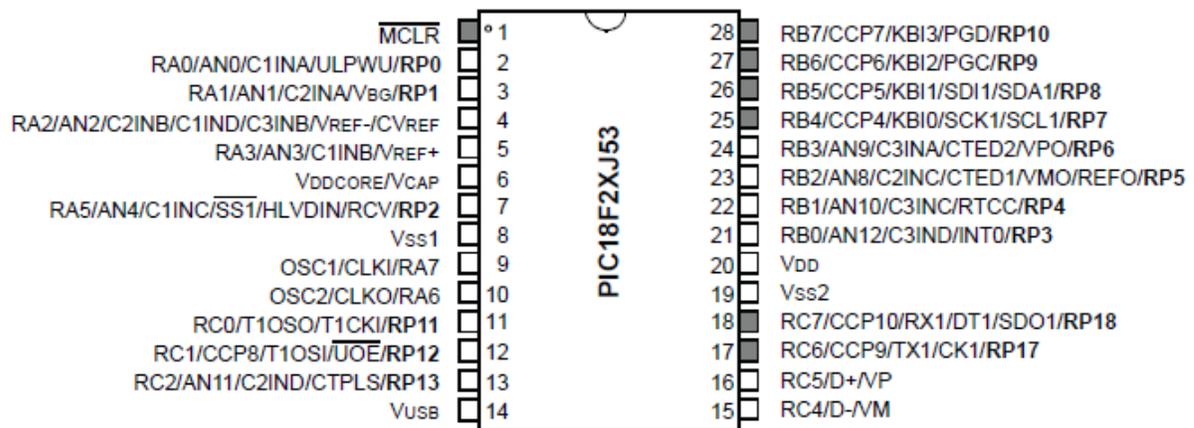


Figura 2.52: Esquema de pines del PIC18F27J53.

Entre algunas de las características de interés se pueden mencionar:

- Tecnología nanoWatt XLP de ultra bajo consumo de potencia especial para aplicaciones con baterías.
- Modo *deep sleep* para disminución de hasta 13nA de consumo. Con capacidad para despertar mediante interrupciones externas.
- Modo *sleep* para disminución de hasta 105nA de consumo. Con capacidad para despertar mediante interrupciones externas.
- Rango de tensión de alimentación: 2.0 a 3.6 V.
- Regulador de tensión interno a 2.5 V.
- Memoria flash de 10KB reescribible con retención de 20 años.
- Módulo USB con soporte para modos *low speed* (1.5Mbps) y *full speed* (12Mbps).
- PLL de alta precisión tanto para el uso del microcontrolador y sus periféricos incluyendo el módulo USB.
- Reloj interno a 31KHz para uso en modos de bajo consumo.
- Reloj interno de alta precisión a 8MHz aumentable mediante la PLL hasta 48MHz. Con capacidad para *boost* de 96MHz durante cortos intervalos de tiempo.
- RTCC interno con capacidad de funcionamiento con el reloj interno del microcontrolador. Capaz de seguir operando aún en modos *sleep* y *deep sleep*. Y disponibilidad de alarmas configurables.
- Dos módulos MSSP para soporte de comunicaciones tipo SPI, 3-Wire e I<sup>2</sup>C.
- Dos módulos USART para soporte de comunicaciones tipo RS-232, RS-485 y LIN/J2602.

- Módulo conversor AD configurable para 10 y 12bits.
- Medición de nivel de tensión de batería.

Se muestra en la Figura 2.53, el diagrama de bloques del PIC18F27J53.

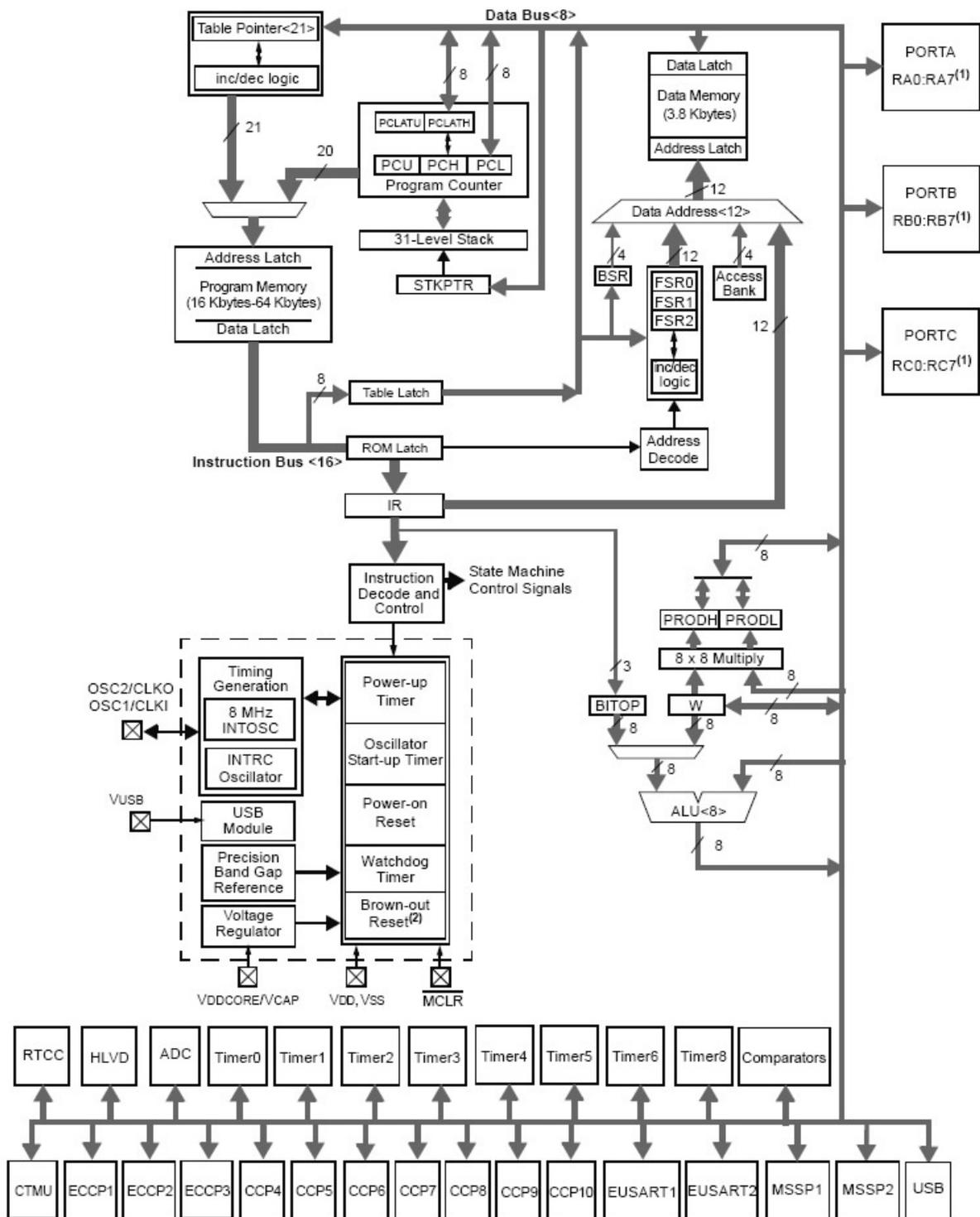


Figura 2.53: Diagrama de Bloques del PIC18F27J53.

# Capítulo 3

## El Marco Metodológico

### 3.1. Tipo de Investigación

Textualmente del trabajo publicado por FEDUPEL a cargo de Maritza Barrios Yaselli, titulado "Manual de Trabajos de Grado de Especialización y Maestrías y Tesis Doctorales"[20] se cita lo siguiente:

"... El Proyecto Factible consiste en la investigación, elaboración y desarrollo de una propuesta de un modelo operativo viable para solucionar problemas, requerimientos o necesidades de organizaciones o grupos sociales; puede referirse a la formulación de políticas, programas, tecnologías, métodos o procesos..."

"... El Proyecto Factible comprende las siguientes etapas generales: diagnóstico, planteamiento y fundamentación teórica de la propuesta; procedimiento metodológico, actividades y recursos necesarios para su ejecución; análisis y conclusiones sobre la viabilidad y realización del Proyecto; y en caso de su desarrollo, la ejecución de la propuesta y la evaluación tanto del proceso como de sus resultados..."

En vista de que el este trabajo se basa en el desarrollo de una red inalámbrica de sensores inteligentes, con la finalidad de contribuir con la situación planteada en el planteamiento del problema del presente trabajo especial de grado, además de coincidir en las etapas a seguir para su desarrollo, este entra dentro del marco expuesto como Proyecto Factible.

### 3.2. Diseño de la Investigación

Textualmente del trabajo publicado por CIEA-Sypal a cargo de Yacqueline Hurtado de Barrera, titulado "Metodología de la Investigación - Guía para la Comprensión Holística de la Ciencia"[21] se cita lo siguiente:

“El diseño de investigación hace explícitos los aspectos operativos del estudio, se define con base en el procedimiento, en este sentido, se refiere a donde y cuando se recopila la información, de modo que se pueda dar respuesta a la situación objeto de investigación de la forma más idónea posible”.

El “donde” del diseño alude a las fuentes: si son vivas, y la información se recoge en su ambiente natural, el diseño se denomina de campo, por el contrario si las fuentes no son vivas, sino documentos o restos, el diseño es documental. También pueden utilizarse diseños de fuente mixta, los cuales abarcan tanto fuentes vivas como documentales. [21]

Para poder cumplir con el objetivo principal de este trabajo y en función de la modalidad de investigación establecida (proyecto factible), el diseño de la presente investigación se considera de fuente mixta, ya que en primer lugar fue necesario la observación de la problemática identificada; y en segundo lugar se realizó una recopilación de información de fuentes bibliográficas, para establecer los fundamentos teóricos relacionados al funcionamiento de los dispositivos compatibles con el estándar IEEE 802.15.4, así como también la información necesaria relacionada al diseño de los módulos.

### **3.3. Técnicas e instrumentos de recolección de datos**

En función de los objetivos planteados, los datos necesarios para la elaboración de este proyecto fueron recolectados en 3 etapas:

- a) Etapa I - Observación directa de la problemática identificada.
- b) Etapa II - Búsqueda y revisión de la documentación disponible entre libros, artículos, normas, especificaciones técnicas, entre otros; relacionado a los módulos de comunicación inalámbrica, sensores de temperatura y presión; además de la socavación de información sobre sensores inteligentes y diseño de redes inalámbricas para la elaboración de un marco bibliográfico que soporte este proyecto especial de grado.
- c) Etapa III - Recolección de datos experimentales siguiendo un protocolo de pruebas para verificar el correcto funcionamiento de todas las funciones asociadas a los dispositivos a construidos.

### **3.4. Procedimiento Metodológico**

En función al cumplimiento de los objetivos propuestos, el trabajo fue desarrollado en seis (6) fases, las cuales se describen a continuación.

#### **3.4.1. Fase I - Descripción del estándar IEEE 802.15.4 y selección del protocolo.**

Esta fase tiene como objetivo la descripción la capa Física (PHY) y la de Control de Acceso al Medio (MAC) del estándar IEEE 802.15.4 para la comprensión del funcionamiento de los

dispositivos compatibles con dicho estándar, además de la selección del protocolo encargado de proveer de la capa de Red y la de Aplicación.

Para cubrir dicho objetivo se llevaron a cabo las actividades mostradas en el Cuadro 3.1.

Cuadro 3.1: Actividades para la descripción de las características del estándar IEEE 802.15.4 y la selección del protocolo para las capas superiores.

Actividades	Descripción de las Actividades
Búsqueda y revisión de la documentación disponible del estándar IEEE 802.15.4	Se realizó la búsqueda del contenido bibliográfico consultando diferentes fuentes, centrandó la atención en la capa Física (PHY) y la capa de Control de Acceso al Medio (MAC) reguladas por el estándar IEEE 802.15.4.
Elaboración de un resumen de las características básicas del estándar.	Se elaboró un resumen de los aspectos resaltantes encontrados en el análisis del estándar en estudio.
Búsqueda y revisión de la documentación disponible de las características de cada protocolo soportado por los módulos de comunicación inalámbrica.	Se realizó la búsqueda del material bibliográfico, consultando variedad de fuentes centrandó la atención en las características de los protocolos MIWI, MIWI P2P y ZigBee.
Elaboración de un cuadro comparativo de las características de los protocolos soportados por los módulos de comunicación inalámbrica compatibles con el estándar IEEE 802.15.4	Se elaboró un cuadro comparativo a partir de los datos obtenidos para contrastar diferencias entre los protocolos y finalmente realizar la selección.

### 3.4.2. Fase II - Diseño del hardware correspondiente a los módulos transceptores.

Esta fase tiene como finalidad el diseño del hardware correspondiente a los módulos transceptores, seleccionando los componentes de menor consumo de potencia con la finalidad de brindarle a dichos módulos un desempeño óptimo dentro de los requerimientos del entorno industrial y optimizar su rendimiento energético.

Para cubrir el objetivo de esta fase se han realizaron las actividades mostradas en el Cuadro 3.2.

Cuadro 3.2: Descripción de las actividades realizadas para el diseño y fabricación de los dispositivos.

Actividades	Descripción de las Actividades
Diseño de los prototipos de dispositivos a implementar.	Se diseñaron los dispositivos correspondientes a: nodo principal de la red, nodos secundarios, sensores de temperatura, sensor de presión y tarjeta de puebas incluyendo circuitos de carga de baterías y circuitos de rectificación de la alimentación de entrada.
Adquisición de los componentes y materiales.	Se listaron los componentes y materiales necesarios para la fabricación de los dispositivos diseñados en la actividad anterior con motivo de analizar su costo y realizar su compra.
Fabricación de los dispositivos descritos.	Se fabricaron los prototipos correspondientes a: nodo principal de la red, nodos secundarios, sensores de temperatura, sensor de presión y tarjetas de puebas.
Programación de los dispositivos.	Se programaron los dispositivos con el código desarrollado para este trabajo especial de grado.

### 3.4.3. Fase III - Implementación de un protocolo de transmisión de paquetes.

Esta fase tiene como objetivo el definir todos los aspectos correspondientes a la trama de datos que conforman el protocolo de transmisión de paquetes de comunicación que permite el enlace entre los nodos de la red inalámbrica además de la incorporación del código de verificación por redundancia cíclica para comprobar la correcta transmisión de los datos de cada comunicación.

Para completar los objetivos de esta fase procederá siguiendo las actividades mostradas en el Cuadro 3.3.

Cuadro 3.3: Descripción de las actividades realizadas para el desarrollo del protocolo de la capa de aplicación.

Actividades	Descripción de las Actividades
Armado de la estructura básica de transferencia para la capa de aplicación.	Se diseñó un formato básico de estructuración para la capa de aplicación que permita el envío de datos y comandos de forma práctica y sencilla.
Creación de los comandos del protocolo	Se creó una lista de comando que permita la configuración de los parámetros de la red y la solicitud de cualquier tipo de información de interés que esta maneje.

### 3.4.4. Fase IV - Desarrollo del algoritmo de la red inteligente.

En esta fase se cubrieron los aspectos relacionados al desarrollo del código del microcontrolador para el manejo de la red inalámbrica. Para ello se procedió según el esquema de actividades en el Cuadro 3.4.

Cuadro 3.4: Descripción de las actividades realizadas para el desarrollo del algoritmo del microcontrolador.

Actividades	Descripción de las Actividades
Desarrollo del algoritmo de las capas inferiores (PHY y MAC)	Se desarrolló un algoritmo bajo las regulaciones establecidas por el estándar IEEE 802.15.4 que provee a todos los dispositivos desarrollados dentro de este trabajo de grado, de las capas PHY y MAC. Este algoritmo permite que un nodo pueda asociarse a la red.
Desarrollo del algoritmo de la capa de red	Se desarrolló un algoritmo basado en las especificaciones del protocolo ZigBee para el desarrollo de la capa de Red que le permite, a cualquier nodo de la red, la transmisión de paquetes de manera inteligente.
Desarrollo del algoritmo para el manejo de la capa de aplicación	Se desarrolló un algoritmo que permite el manejo del protocolo diseñado en este trabajo de grado para la capa de aplicación.

### 3.4.5. Fase V - Desarrollar un software de aplicación Windows™ con interfaz gráfica.

Esta fase tiene como objetivo el desarrollo de la interfaz de usuario que permite la configuración de los parámetros de la red, además de la medición en tiempo real de parámetros relativos al estado del sensor. Se muestran en el Cuadro , la descripción de las actividades pertinentes a su desarrollo.

Cuadro 3.5: Descripción de las actividades realizadas para el desarrollo de la interfaz gráfica de usuario.

Actividades	Descripción de las Actividades
Desarrollo de la interfaz gráfica	Mediante el uso de Visual Studio 2010, se desarrolló una interfaz gráfica que permite la configuración de los parámetros de la red además de la visualización de parámetros de medición como el TAG del sensor, factor de calibración, unidad de medición, entre otros.

### 3.4.6. Fase VI - Verificación de correcto funcionamiento.

Esta fase tiene como objetivo la comprobación del funcionamiento en general, en todos sus aspectos, de los dispositivos y la interfaz.

#### 3.4.6.1. Pruebas de Comunicación

Las pruebas a desarrollar a continuación fueron realizadas en un solo proceso por lo que el estado final de cada prueba equivale al estado inicial de la siguiente, a menos que se indique lo contrario.

**Comunicación: Interfaz - Nodo Principal** Esta prueba consistió en:

- Instalar la interfaz en el computador.
- Abrir la interfaz.
- Conectar el dispositivo (Nodo principal) al computador.
- Verificar el reconocimiento del dispositivo por parte de la interfaz.
- Envío del comando de *reset* dirigido al nodo principal.
- Verificación del reinicio del nodo principal (Detección de desconexión y reconexión a la interfaz).

**Proceso de Asociación: Nodo Principal - Nodo** Después de haber verificado la comunicación del nodo principal con la Interfaz se realizaron las pruebas contenidas en el siguiente listado:

- Abrir el software desarrollado para depuración de errores en los procesos de la red (Debugger).
- Activación de un nodo secundario dentro del alcance del nodo principal utilizando como alimentación un adaptador AC/DC a 9V con una capacidad de suministro mayor a 100mA.
- Verificación de la recepción y transmisión de todos los paquetes correspondientes al proceso de asociación.
- Verificación de correcta asociación del dispositivo recientemente añadido a la red mediante la solicitud de su lista de nodos cercanos, solicitud de rango de asignación para futuras asignaciones Nodo-Nodo, observación de los parámetros asociados a la red como: Canal, potencia de antena, entre otros.

**Comunicación: Interfaz - Nodo Principal - Nodo** Se realizaron las siguientes pruebas para verificar la comunicación con el nodo secundario

- Envío del comando de *reset* al nuevo dispositivo desde el computador.
- Verificación de desconexión y reconexión al nodo principal bajo los mismo parámetros que le fueron asignados la vez anterior.
- Solicitud de mediciones realizadas por el nodo.
- Verificación de correcta medición.

**Proceso de Asociación: Nodo - Nodo** Para la verificación del correcto funcionamiento de los procesos de asociación nodo-nodo se realizaron las siguientes pruebas:

- Haciendo uso de un segundo computador en el cual se abrió el software de depuración de errores (Debugger).
- Se conectó a este segundo computador un segundo nodo secundario el cual estuviera dentro del alcance del nodo secundario que ya había sido asociado a la red y fuera del alcance del nodo principal.
- Se verificó la recepción y transmisión de todos los paquetes requeridos en el proceso de asociación Nodo-Nodo en el Debugger.
- Verificación, desde la interfaz, la correcta asociación del dispositivo recientemente añadido a la red mediante la solicitud de su lista de nodos cercanos, solicitud de rango de asignación para futuras asignaciones Nodo-Nodo, observación de los parámetros asociados a la red como: canal, potencia de antena, entre otros.

**Comunicación: Interfaz - Nodo Principal - Nodo Puente - Nodo** Para la verificación del correcto funcionamiento de los procesos de comunicación desde la interfaz al nuevo nodo secundario, se realizaron las siguientes pruebas:

- Se cambia la tarjeta de pruebas y se le conecta la tarjeta para la medición de temperatura.
- Envío del comando de *reset* al nuevo dispositivo desde el computador.
- Verificación de desconexión y reconexión al nodo secundario al que estaba enlazado bajo los mismo parámetros que le fueron asignados la vez anterior.
- Verificación de correcta medición.

**Enrutamiento y listado de errores** Esta prueba, rompiendo con la línea de pruebas anterior, se realizó en un escenario donde un par de nodos secundarios se asocian a la red y un tercero a alguno de estos últimos. Para la prueba de enrutamiento se hizo lo siguiente:

- Se verificó el correcto envío y recepción de datos al tercer nodo desde la interfaz.
- Se solicitaron listas de rutas para los 3 nodos secundarios.
- Se observó cual de los nodos intermedios estaba funcionando de nodo puente.
- Se inhabilitó el nodo detectado como nodo puente.
- Se envió desde la interfaz una solicitud de medición.
- Se verificó la correcta recepción de la solicitud por parte del tercer nodo.
- Se solicitaron listas de rutas y de errores para el par de nodos secundarios restantes.
- Se verificaron tanto el error listado como la nueva ruta creada.

#### **3.4.6.2. Pruebas del Protocolo de la Capa de Aplicación**

Se verificó el correcto funcionamiento del protocolo de comunicación para la capa de aplicación de la siguiente forma:

- Se ejecutaron todos los comandos pertenecientes al protocolo de comunicaciones de la capa de aplicación entre la interfaz y el nodo principal y se verificó su correcto funcionamiento a través del debugger.
- Se ejecutaron todos los comandos pertenecientes al protocolo de comunicaciones de la capa de aplicación entre la interfaz y un nodo secundario y se verificó su correcto funcionamiento a través del debugger.
- Se ejecutaron todos los comandos pertenecientes al protocolo de comunicaciones de la capa de aplicación entre la interfaz y un nodo secundario haciendo uso de un nodo intermedio y se verificó su correcto funcionamiento a través del debugger.

# Capítulo 4

## Los Resultados y su Análisis

Es presentan a continuación los resultados obtenidos de cada una de las fases planteadas en el Marco Metodológico.

### 4.1. Descripción del Estándar y la Selección del Protocolo

Esta sección tiene como objetivo la descripción de la Capa Físicas (PHY) y la Capa de Control de Acceso al Medio (MAC) del Estándar IEEE 802.15.4. además de la selección del protocolo de la capa de red.

#### 4.1.1. Resumen del Estándar IEEE 802.15.4

La capa física es la responsable de regular todos los aspectos concernientes al canal, como definición del medio de comunicación, bandas de frecuencia, control de activación y desactivación de módulos de emisión y recepción, entre otros; y la sincronización de todos los dispositivos en la red a la hora de establecer una comunicación. Esta compuesta por tres(3) componentes: El Encabezado de Sincronización (SHR) encargado de manejar el preámbulo de sincronización, El Encabezado de la Capa Física (PHR) que indica la longitud de su Carga y la trama de Carga de la Capa Física (PHY Payload) encarga de contener a todas las capas superiores.

La capa MAC, en cambio, es la encargada de regular los procesos de asociación y disociación de la red, realiza todos estos procesos bajo las tramas *beacon frame*, *command frame*, *data frame* y *acknowledge frame*. Está compuesta de igual forma por 3 componentes: El encabezado de la capa MAC (MHR) encargada de contener los identificadores de red e identificadores de dispositivos de origen y destino de cada paquete además de contener información de la encriptación de la carga, entre otros; La carga de la MAC (MAC Payload) encargada de contener las capas superiores; y el pie de MAC (MFR) encargada de contener los bytes de FCS (CRC de 16 bits de la ITU).

Dentro de las topologías que soporta el estándar IEEE 802.15.4 se pueden mencionar: estrella, peer-to-peer y mesh.

### 4.1.2. La selección del protocolo de la capa de red

Se estudiaron los tres protocolos soportados por los módulos compatibles con el estándar IEEE 802.15.4, analizados dentro de este trabajo de grado. Con la información recopilada se armó el Cuadro 4.1.

Cuadro 4.1: Comparación entre los protocolos MIWI, MIWI P2P y ZigBee.

Característica\Protocolo	MIWI P2P	MIWI	ZigBee
Estándar	Microchip	Microchip	Estándar Industrial
Tamaño de la Red	Directa	Pequeña	Muy Grande
Cantidad de Hops	1	4	Ilimitados
Cantidad de Nodos	$2^{64}$	~1K	~64K
Req. Coord. de PAN	Si	Si	Si
Número de Coord.	No admite	8	Ilimitados
Req. Mem. No Volátil	Opcional	Opcional	~2Kb
Costo	0	0	Mayor a 9000\$
Certificación	Ninguna	Ninguna	ZigBee Alliance
Gamma que lo Soporta	18,24,30,32 y 33	18,24,30,32 y 33	24,30,32 y 33
Tamaño Stack	~4Kb	~16Kb	64-128Kb

Por brindar un amplio número de nodos en la red, sin límite de coordinadores, disponibilidad para redes muy grandes con capacidad para comunicar un paquete a través de enlaces con infinidad de nodos intermedios y por estar soportado por un estándar libre; esto aunado a su alto costo, llevó a la selección de la opción del desarrollo de un código basado en ZigBee, para la capa de Red, que permita todas estas facultades logrando un código superior al brindado por MIWI y MIWI P2P.

## 4.2. Diseño y Fabricación del Hardware

Esta sección describe el proceso de diseño y fabricación del hardware correspondiente a los módulos transeptores que conforman la red inalámbrica. Para ello se desglosaron los procesos en dos tramos: el diseño de los dispositivos y la fabricación de los dispositivos los cuales se describen a continuación.

### 4.2.1. Diseño de los Dispositivos

#### 4.2.1.1. Selección del Módulo de Comunicación Inalámbrica

Para la selección del módulo de comunicaciones se elaboró el Cuadro 4.2, a fin de poder comparar las características de los módulos estudiados en este trabajo de grado y facilitar su selección.

Cuadro 4.2: Cuadro comparativa de los módulos de comunicación inalámbrica compatibles con el estándar IEEE 802.15.4 bajo estudio.

Característica\Módulo	ZigBit	Xbee	XBeePro	ProFLEX01	MRF24J40MA	MRF24J40MB
Dimensiones [mm]	24x13.5x2.0	24.38x27.61x2.79	24.38x32.94x2.79	22.86x41.40x5.08	17.78x27.94.40x1.70	22.86x33.02x3.35
Rango de Alimentación [V]	1.8 a 3.6	2.8 a 3.4	2.8 a 3.4	2.2 a 3.3	2.4 a 3.6	2.4 a 3.6
Consumo Tx [mA]	18	45	150	145	23	130
Consumo Rx [mA]	19	50	180	30	19	25
Modo Bajo Consumo [uA]	6	10	10	8	2	5
Canales	16	16	16	16	16	16
Rango de Potencia [dBm]	-17 a 3	-10 a 0	-3 a 10	7 a 20	-36 a 0	-36 a 20
Sensibilidad de Antena [dBm]	-101	-92	-100	-98	-94	-102
Ancho de Banda [Kbps]	250	250	250	250	250 y 650	250 y 650
Interfaz de Comunicación	ZigBee	ZigBee	ZigBee	ZigBee	ZigBee y Protocolos Personales	ZigBee y Protocolos Personales
Protocolos Soportados	ZigBee	ZigBee	ZigBee	ZigBee	MIWI, MIWI P2p, ZigBee y Protocolos Personales	MIWI, MIWI P2p, ZigBee y Protocolos Personales
Manejo de Topología Mesh	Si	Si	Si	No	Si	Si
Costo [Dólares USA]	54.54	28.95	56.96	22.49	9.12	17.82

Se seleccionó el módulo de comunicación inalámbrica MRF24J40MA debido a que es el que posee el más bajo costo, soporte para protocolos personales y ZigBee, que compensa su baja potencia de transmisión ya que estos permiten procesos multi-saltos sin límite que permiten que el tamaño de la red pueda extenderse enormemente y cubrir grandes áreas, por ser el segundo en menor consumo de potencia, ser el segundo más compacto además de que soporta la topología mesh.

#### 4.2.1.2. Selección del Microcontrolador

Como requerimientos básicos necesarios del microcontrolador que se seleccionó se tienen:

- Bajo consumo de potencia.
- Rango de operación cercano a los 3.3V.
- Soporte para comunicación USB.
- Soporte para comunicación SPI.
- Conversor Analógico Digital de 12 bits.
- RTCC interno para la identificación en tiempo de errores a la hora de ser listados.
- Memoria disponible para almacenamiento de datos.

Estos parámetros se introdujeron en el buscador avanzado de partes de la Microchip "MAPS"[22] arrojando como resultado una serie de microcontroladores entre los cuales se seleccionó el PIC18F27J53 por cumplir con todos los requerimientos y poseer la menor cantidad de terminales.

#### 4.2.1.3. Diseño del Circuito de Rectificación

Por motivo de protección se diseñó un circuito de protección que permita alimentar los dispositivos tanto con señales de corriente alterna como de corriente directa esto mediante un puente rectificador de onda completa y un circuito de regulación de tensión como se muestra en la Figura 4.1. En dicha figura se observa de igual forma que la salida de alimentación y tierra tiene conexión con el puerto de conexiones como lo indican los señaladores PIN\_OUT\_Vin y PIN\_OUT\_GND respectivamente. La señal V1 simula la señal de entrada proveniente del Jack de alimentación y se toman de los puntos Vin y GND la tensión de entrada para el circuito de carga de baterías y el de regulación de alimentación a 3.3V.

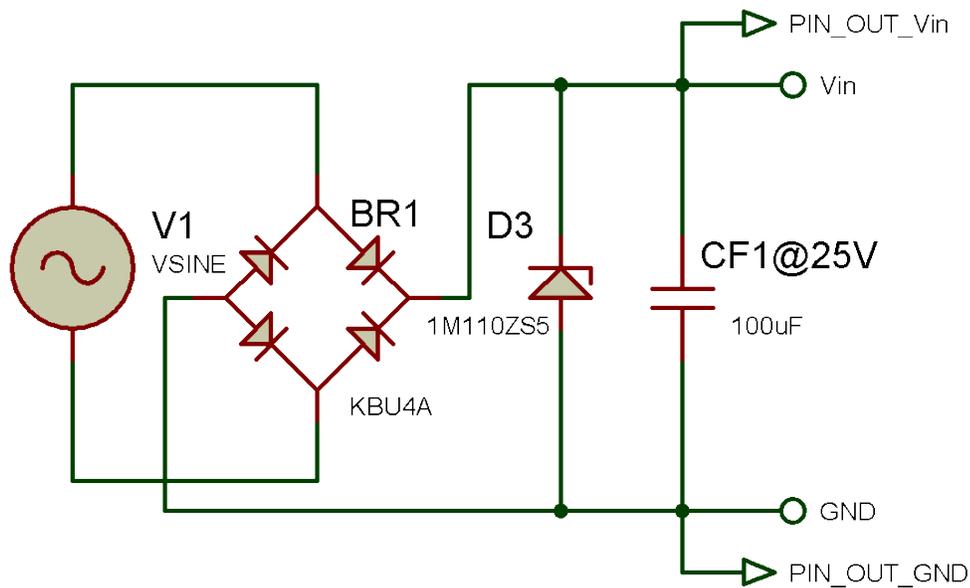


Figura 4.1: Circuito de rectificación de la tensión de alimentación.

#### 4.2.1.4. Diseño del Circuito de Carga de Batería

Se diseñó este circuito como una modificación del regulador de tensión limitado en corriente expuesto en la hoja de especificaciones del regulador LM317[23] al cual se le fue añadido un circuito de regulación de tensión a 3.3V mediante el uso de un regulador de tensión NTE1904, como se muestra en la Figura 4.2, dejando como opcional la utilización de diodos de protección de corriente inversa al NTE1904. En dicha figura se marca en rojo la resistencia ajustable de calibración para regular la tensión de carga de baterías.

#### 4.2.1.5. Consideraciones Varias

- Se tomaron consideraciones de protección de sobre-corriente para los pines INT y RESET del módulo de comunicación inalámbrica mediante el uso de resistencias en serie de 1K.
- Para protección de los pines restantes del microcontrolador se colocaron resistencias en serie de 100 Ohm y 1/8W para prevenir averías en pines del microcontrolador.
- Se dejaron como opcionales resistencias de pull-up y pull-down para los pines de comunicación del MRF24J40MA.
- Se colocaron capacitores de 0.1µF a la entrada de alimentación del PIC18F27J53 y del convertor AD diferencial, para disminución de ruido electromagnético a la entrada de cada uno.
- Se dejó como opcional un capacitor de 0.1µF a la entrada de alimentación de MRF24J40MA por este contar con un capacitor interno a la entrada de 1µF.
- Se colocaron capacitores de 10µF para la alimentación del módulo USB

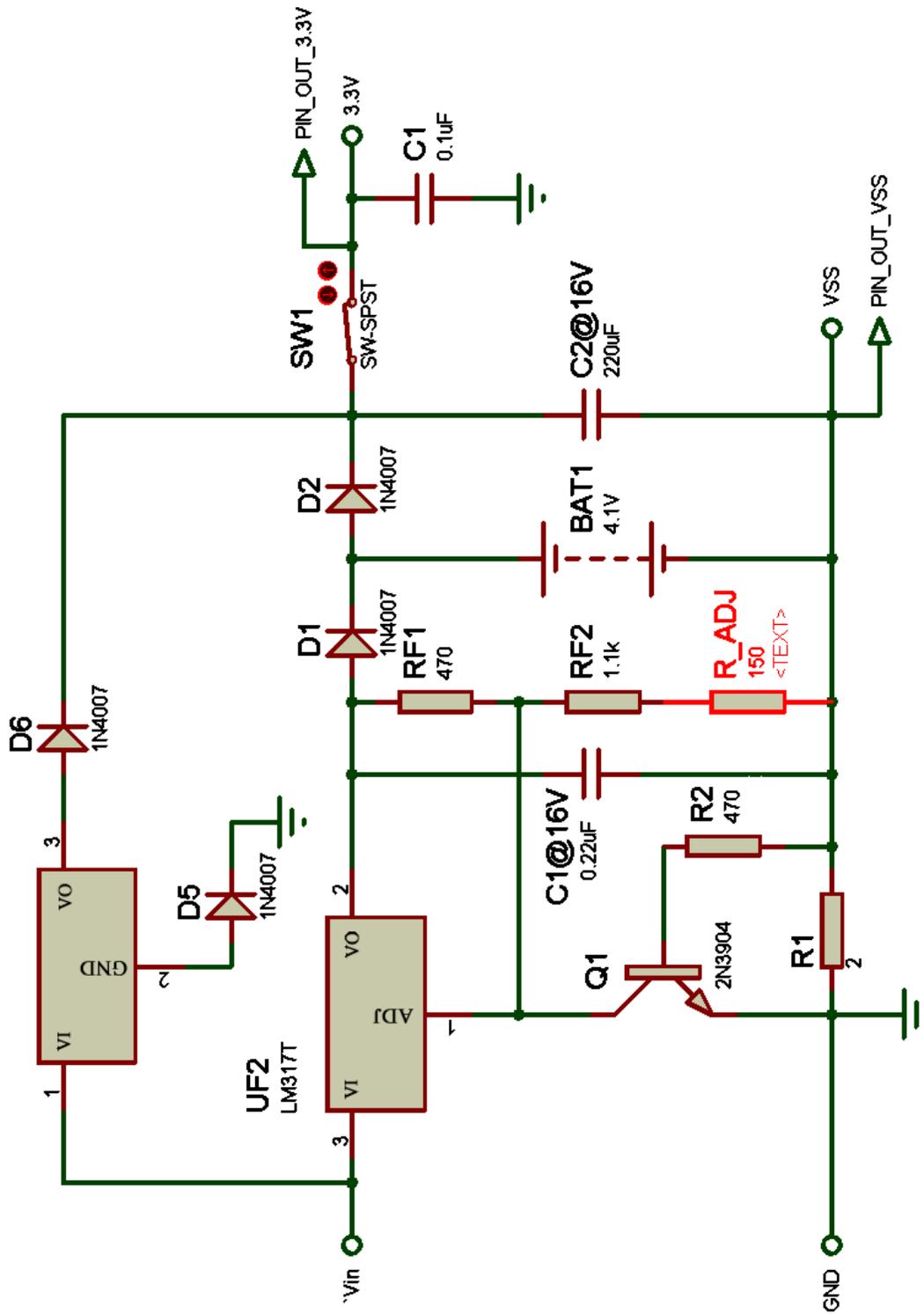


Figura 4.2: Simulación del circuito de Carga de Batería.

#### 4.2.1.6. Diseño Nodo Principal

Este nodo fue diseñado en función al esquema mostrado en la Figura 4.3, se pueden observar los siguientes aspectos:

- Como fuente de alimentación se tiene al puerto USB, la cual es regulada mediante el uso de un NTE1904 a 3.3V, por ser esta la tensión de operación tanto del microcontrolador como del módulo de comunicación inalámbrica.
- La interfaz SPI de comunicación entre el MRF24J40MA y el PIC18F27J53 y el led indicador de transferencia de datos conectado al puerto de entradas y salidas digitales.
- La conexión del terminal de interrupciones (INT) del módulo de comunicación inalámbrica al terminal de interrupciones externas (INT0) para la atención inmediata de cualquier evento ocurrido en el MRF24J40MA.
- El control en software del RESET del MRF24J40MA por parte del microcontrolador a través de un terminal I/O.
- La comunicación entre este nodo y el computador se realizó bajo el protocolo USB.

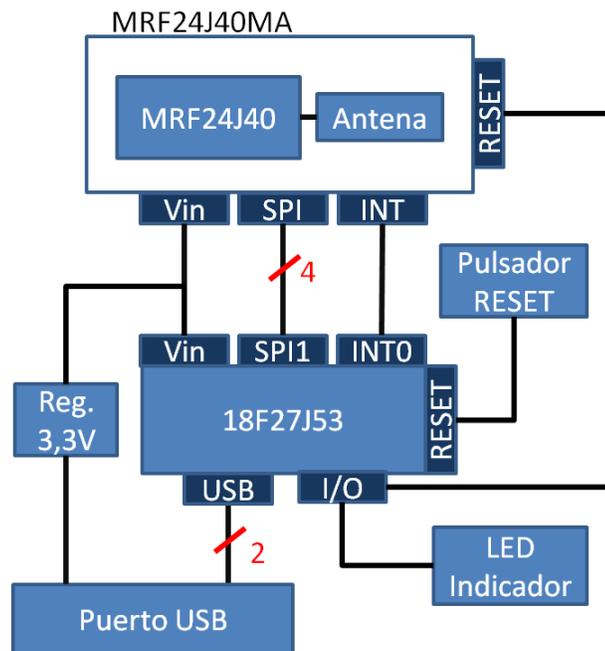


Figura 4.3: Diagrama de bloques del nodo principal de la red.

Entre algunos de los esquemas de conexiones se muestran, en la Figura 4.4, el del PIC18F27J53 donde se observan las salidas al módulo de comunicación inalámbrica denotadas bajo los identificadores MRF, de igual forma las líneas del bus de comunicación USB Data+ y Data- se denotan con el identificador USB y los pines de programación se denotan bajo los identificadores PGRM\_PIN.

Se observa además, el led indicador de transmisión conectado al pin RC6 mediante una resistencia de 1KΩ.

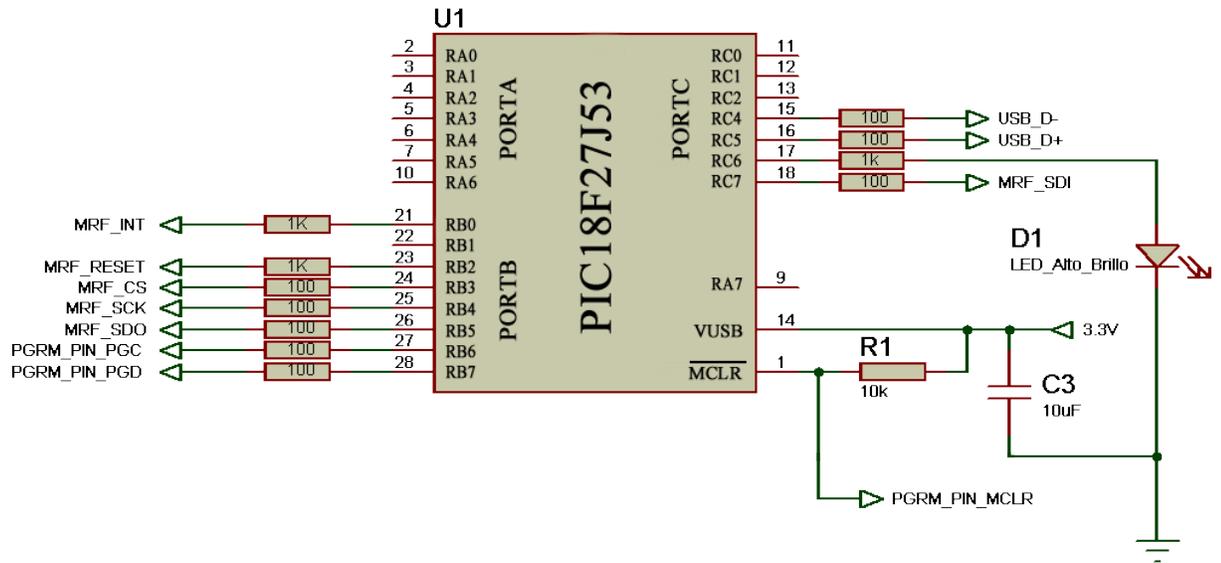


Figura 4.4: Diagrama de conexiones del PIC18F27J53 para el nodo principal de la red.

Otro de los esquemas de interés se muestra en la Figura 4.5, correspondiente al diagrama de conexiones del MRF24J40MA, donde se observan los terminales de conexión a tensión, tierra, pines de la interfaz SPI de comunicación, terminal de interrupción y los terminales de NC (No Connect) y WAKE. Cabe destacar que el pin de WAKE no fue conectado ya que es un terminal que funciona bajo el modo de operación *beacon enable*, modo que no corresponde al modo de operación de esta red.

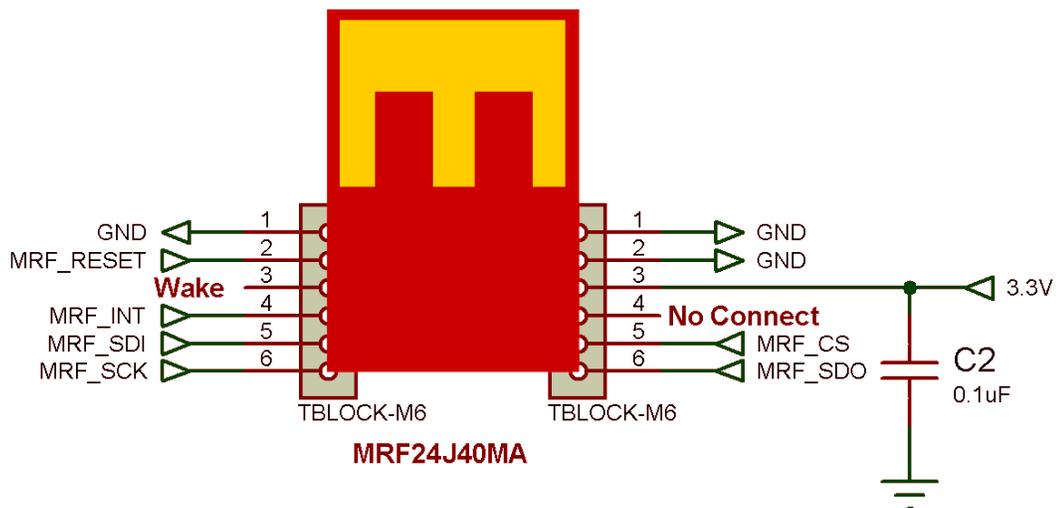


Figura 4.5: Diagrama de conexiones del MRF24J40MA para el nodo principal de la red.

Por último en el diagrama circuital de regulación, el de los terminales de programación y los del puerto USB se muestran en la Figura 4.6. de izquierda a derecha respectivamente.

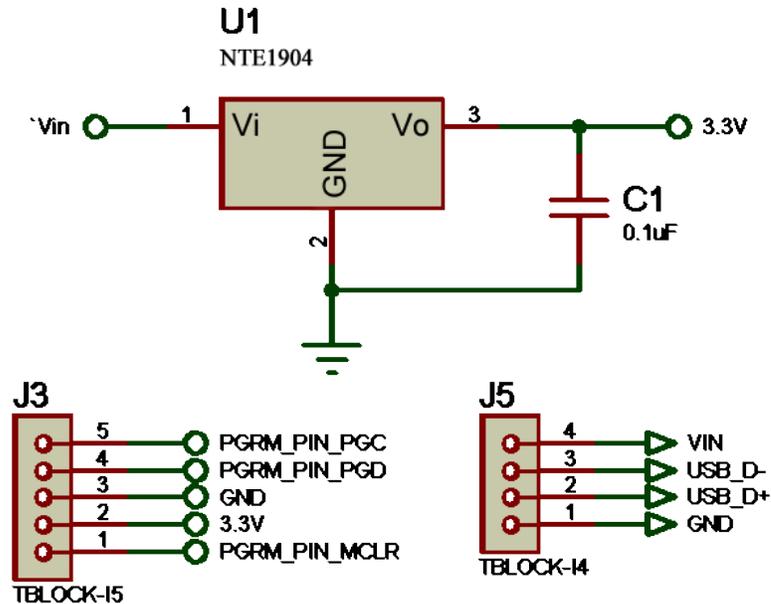


Figura 4.6: Diagramas de alimentación y pines de conexión del nodo principal. Se observa en la parte superior el circuito de regulación a 3.3V y el capacitor de 0.1µF del PIC, abajo a la izquierda los pines de programación y a la derecha los pines del puerto USB.

Haciendo uso del software Eagle v6.3, se diseñó la tarjeta de circuito impreso para el nodo principal como se muestra en la Figura 4.7. En dicha Figura se muestra en color rojo las pistas de la parte anterior de la tarjeta de circuito impreso y en azul las de la parte posterior. Cabe destacar que los componentes fueron diseñado para estar del mismo lado del plano azul.

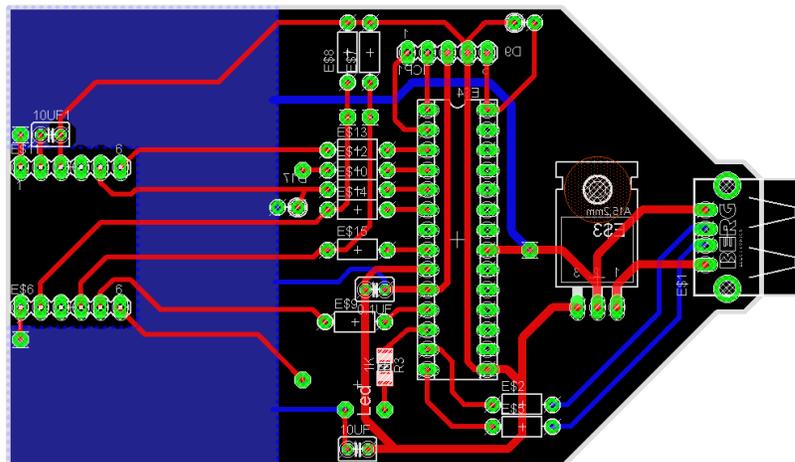


Figura 4.7: Diseño en Eagle de la tarjeta de circuito impreso para el nodo principal.

#### 4.2.1.7. Diseño Nodo Sensor ADC Simple

Este nodo conforma el resto de los nodos de la red y su diseño se rigió bajo el esquema mostrado en la Figura 4.8, donde se muestra:

- Un módulo de alimentación que consta de un jack de alimentación con tensiones de entrada de 9 a 24V AC/DC, el circuito de rectificación, un circuito de regulación de tensión utilizado mientras el dispositivo se encuentra conectado a la toma de energía eléctrica, el circuito de carga de baterías alimentado por la salida del rectificador, las baterías y un interruptor que desacopla el módulo de alimentación de la carga interrumpiendo el suministro de energía del Nodo.
- Un bloque representando al PIC18F27J53 mostrando sus principales conexiones como lo son: el módulo de conversión analógico digital, el módulo SPI para comunicación con el MRF24J40MA, el circuito de interrupciones descrito en el desarrollo del nodo principal, un led indicador de transferencia de datos, un pulsador de *reset* y un puerto de conexiones donde se ubican cuatro(4) canales de conversión analógico digitales, tres(3) canales digitales, alimentación (3.3V), tierra y pines de comunicación USB.

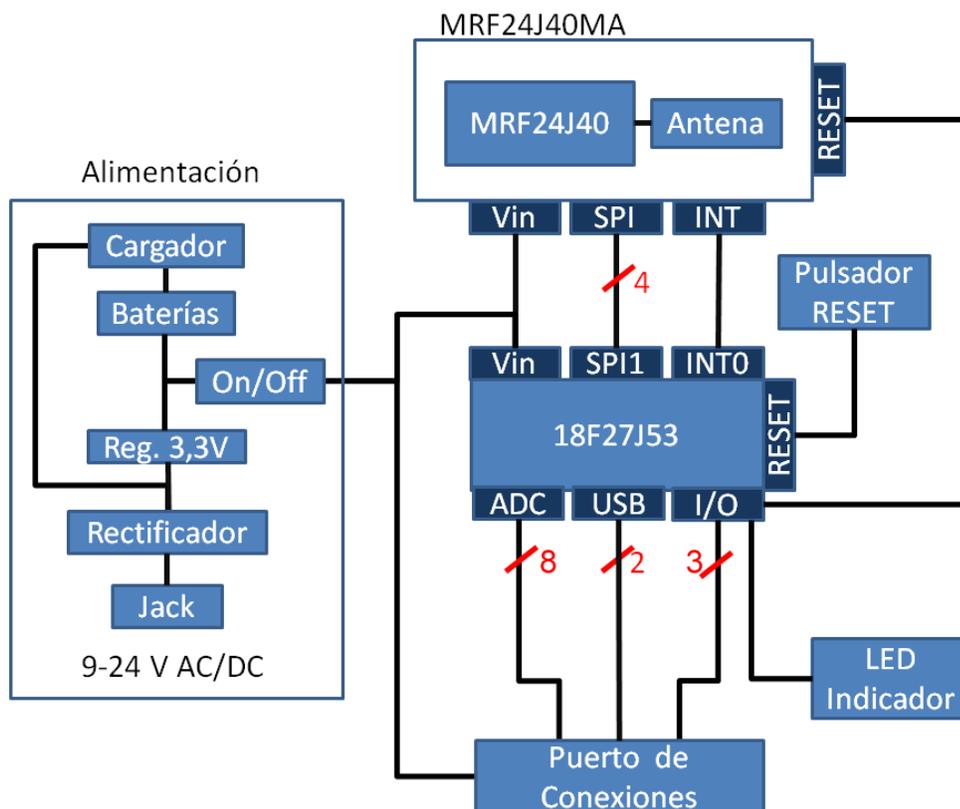


Figura 4.8: Diagrama de bloques de los nodos sensores de la red con ADC simple.

Dentro de los esquemas de conexiones se puede observar en la Figura 4.9, el esquema de conexiones para el PIC18F27J53 para un nodo sensor. De igual forma se muestran los mismos

indicadores PIN\_OUT, MRF, USB y PGRM\_PIN para pines del puerto de conexiones, pines del MRF24J40MA, pines del USB (Data+ y Data-) y pines de programación. A diferencia del nodo principal, este tiene un pulsador de *reset* además del puerto de conexiones. Se observan en la figura mencionada anteriormente algunas de las consideraciones listadas en la subsección 4.2.1.5 como los son las resistencias de protección de 100Ω y 1KΩ y capacitor de 10uF para USB.

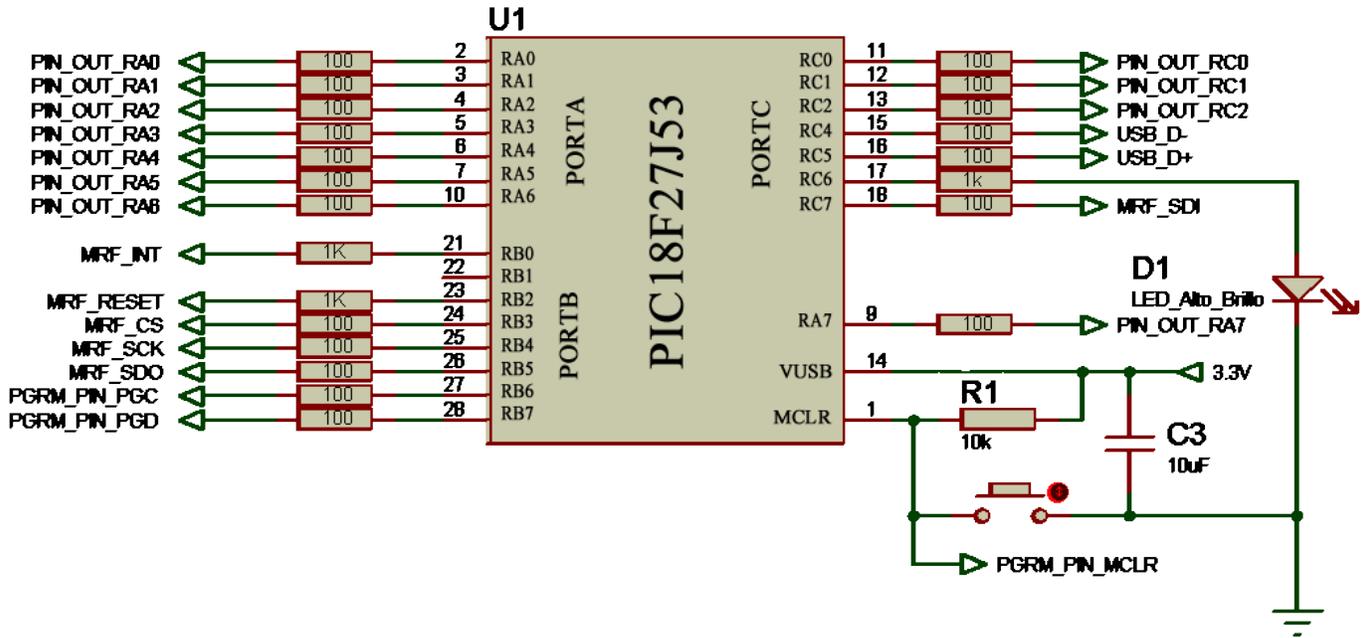


Figura 4.9: Esquema de conexiones del PIC18F27J53 para un nodo sensor.

El esquema de conexiones del MRF24J40MA no varía con respecto al mostrado en la Figura 4.5, por lo tanto solo se hará referencia a esta. Por otra parte este nodo presenta un puerto de conexiones que puede observarse en la Figura 4.10, donde se muestran los terminales a los que se tiene acceso.

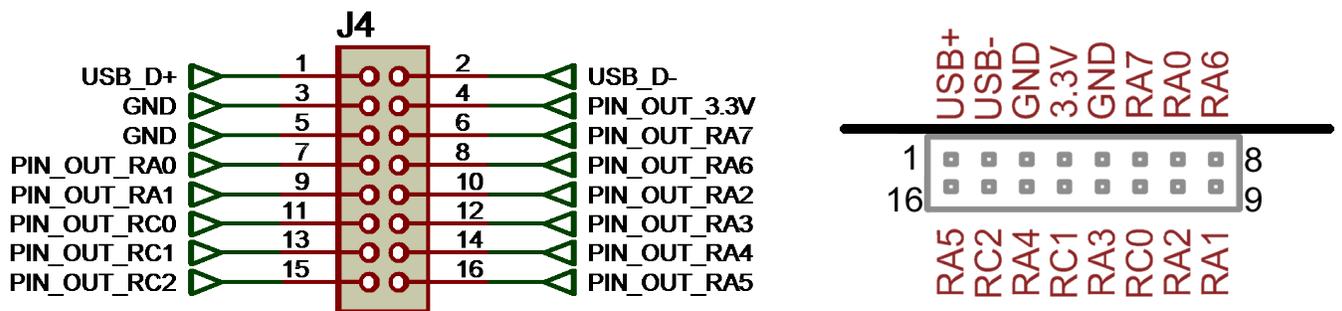


Figura 4.10: Diagramas del puerto de conexiones del nodo sensor con ADC sencillo. A la izquierda se muestra el esquema de conexiones y a la derecha la vista frontal frontal del puerto de conexiones.

Finalmente se diseñó la tarjeta de circuito impreso en el software Eagle como se muestra en la Figura 4.11, donde se observa en rojo la capa anterior y en azul la posterior. Cabe destacar que los componentes van colocados del lado posterior para ser soldados del lado anterior.

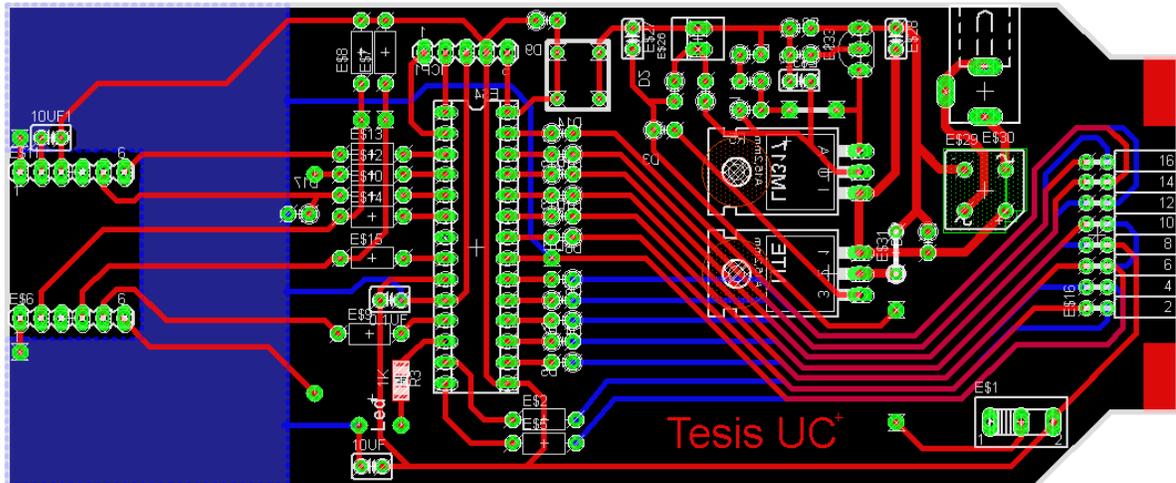


Figura 4.11: Diseño de la tarjeta de circuito impreso para el Nodo Sensor con ADC Simple.

#### 4.2.1.8. Diseño Nodo Sensor ADC Diferencial

Este nodo se realizó bajo el esquema mostrado en la Figura 4.12, en donde se observa una composición similar a la del nodo sensor descrito en la sección 4.2.1.7, teniendo de igual forma un módulo de alimentación, un circuito asociado a la comunicación entre el módulo de comunicación inalámbrica y el microcontrolador, circuito de *reset*, entre otros. La diferencia fundamental surge al integrar el convertor analógico digital de 12 bits MCP3204, que provee de 4 canales simples o 2 canales diferenciales, convertor que se añade por el microcontrolador no contar con dicha medición diferencial. En el puerto de conexiones se sustituyeron los 4 terminales RA5, RC2, RC1 y RC0 por los canales del convertor CH3, CH2, CH1 y CH0 respectivamente. Dicho convertor tiene, al igual que el MRF24J40MA, un módulo SPI como interfaz de conexión para comunicación con otros dispositivos, debido a esto fue necesario hacer uso del segundo módulo SPI del cual dispone el microcontrolador 18F27J53 el cual se implementa mapeando sus pines a los pines indicados como remapeables (RP) del microcontrolador. Dichos terminales fueron dirigidos a los pines RC0, RC1, RC2 y RB1 correspondiente a los terminales Dout, Din, Chip Select y Clk respectivamente.

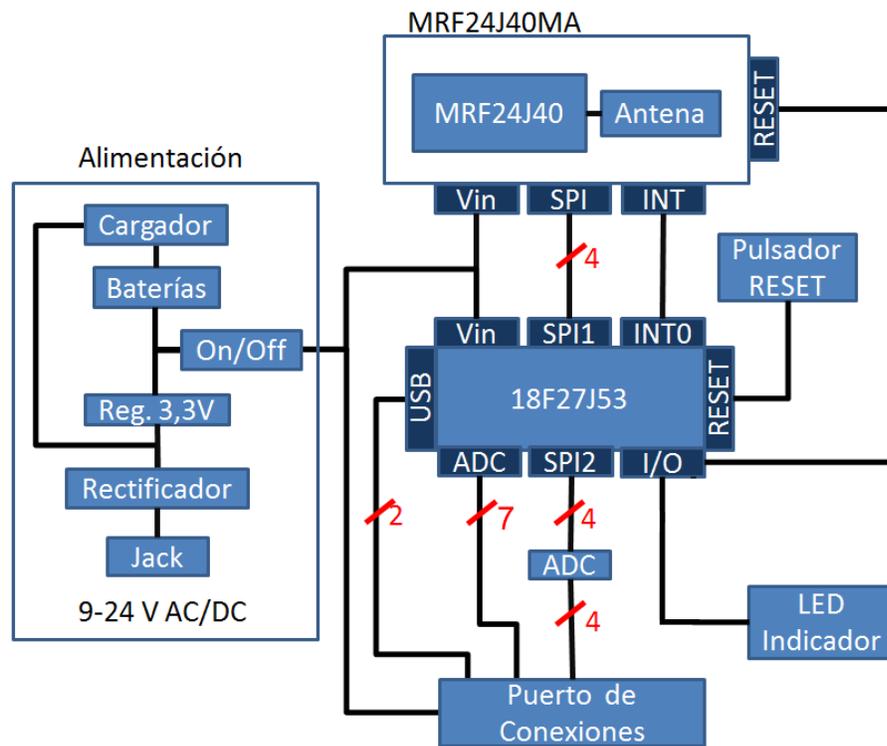


Figura 4.12: Diagrama de Bloques de los nodos sensores de la red con ADC diferencial.

Es de interés entonces mostrar la distribución de terminales resultante del puerto de conexiones al añadir el convertor AD diferencial, el cual se muestra en la Figura 4.13

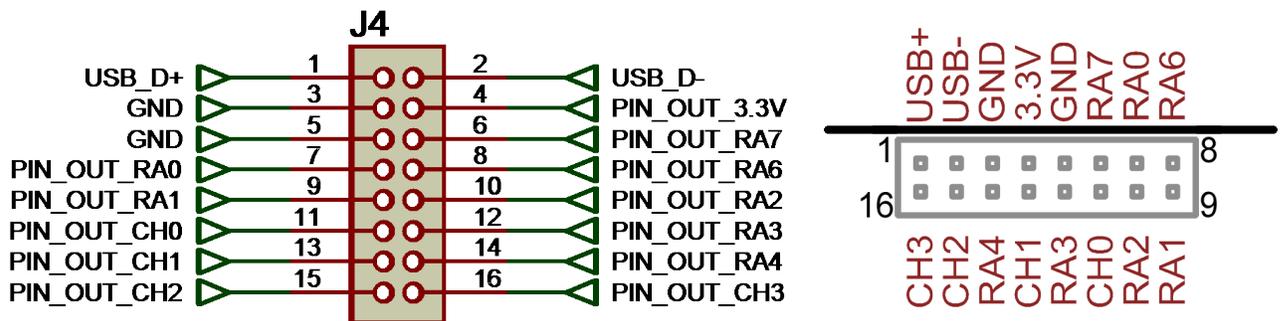


Figura 4.13: Diagramas del puerto de conexiones del nodo sensor con ADC diferencial. A la izquierda se muestra el esquema de conexiones y a la derecha la vista frontal del puerto de conexiones.

De igual forma se hicieron los ajustes para el diseño de la tarjeta de circuito impreso integrando el módulo de conversión analógico digital y dándole acceso a los canales a través del puerto de conexiones. Cabe destacar que este convertor es de tipo SMD y fue montado en la cara posterior (azul) como puede observarse en la Figura 4.14

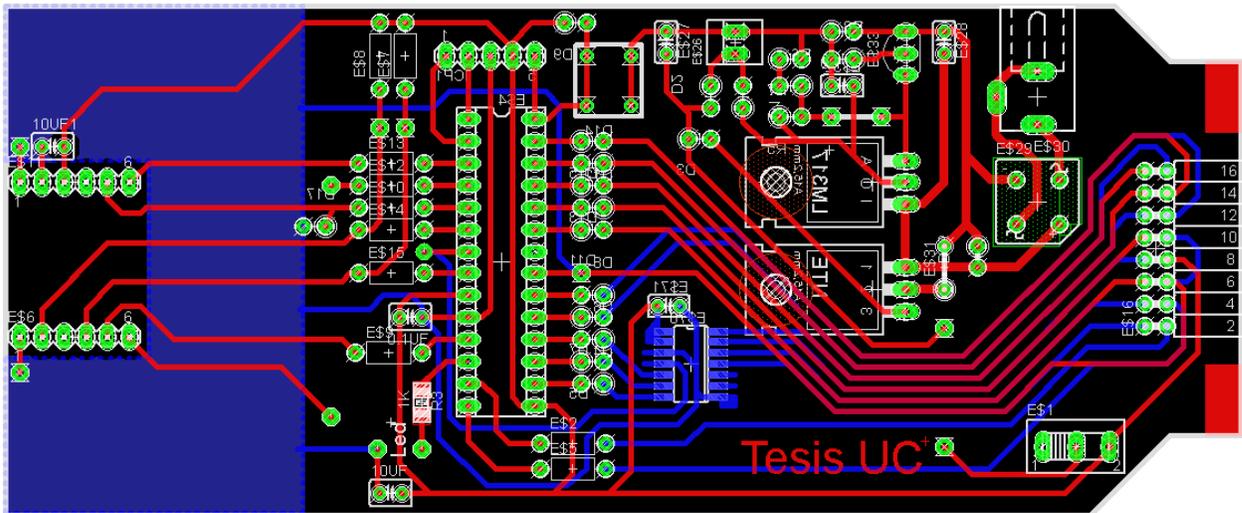


Figura 4.14: Diseño de la tarjeta de circuito impreso para el Nodo Sensor con ADC diferencial.

#### 4.2.1.9. Diseño de Sensor de Temperatura y Presión

Los sensores se diseñaron en base al diagrama mostrado en la Figura 4.15, donde se observa el sensor cuya medición pasa por un proceso de filtrado y finalmente al convertor analógico digital, ya sea el del microcontrolador o el integrado, para mediciones simples o diferenciales respectivamente.

Se utilizó como sensor de presión un MPX53DP, empaquetado 344C con salida diferencial y como sensor de temperatura un MCP9700A ambos por admitir la misma tensión de operación del microcontrolador y del módulo de comunicación inalámbrica (3.3V).

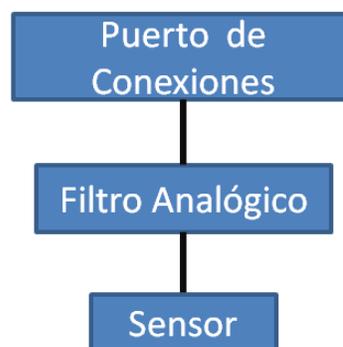


Figura 4.15: Diagrama de Bloques de los Nodos Sensores de la Red.

El cálculo del filtro se realizó mediante un circuito RC Serie cuya ecuación de frecuencia de corte corresponde con:

$$f_o = \frac{1}{2.\pi.R.C}$$

Fijando el valor de la capacitancia  $C = 100\mu F$ , tomando como criterio una frecuencia de corte mucho menor a 60Hz, como por ejemplo 0.08Hz en vista de que la pendiente del filtro es de 20dB por década, con lo que la atenuación a los 60Hz sería de aproximadamente 60dB, suficiente como para considerar la interferencia a dicha frecuencia anulada; y teniendo en cuenta que la impedancia de entrada típica para un canal de conversión analógico digital es de aproximadamente 10K, estando esta última, en serie con nuestra resistencia de filtrado, es decir,  $R = R_{FILTRO} + R_{ADC}$ . Sustituyendo y despejando  $R_{FILTRO}$  se obtiene que:

$$R_{FILTRO} \approx 9894,37 \approx 10K$$

Los esquemas circuitales de los filtros se pueden observar en la Figura 4.16, en donde se encuentra el filtro simple a la izquierda y el filtro para la señal diferencial a la derecha.

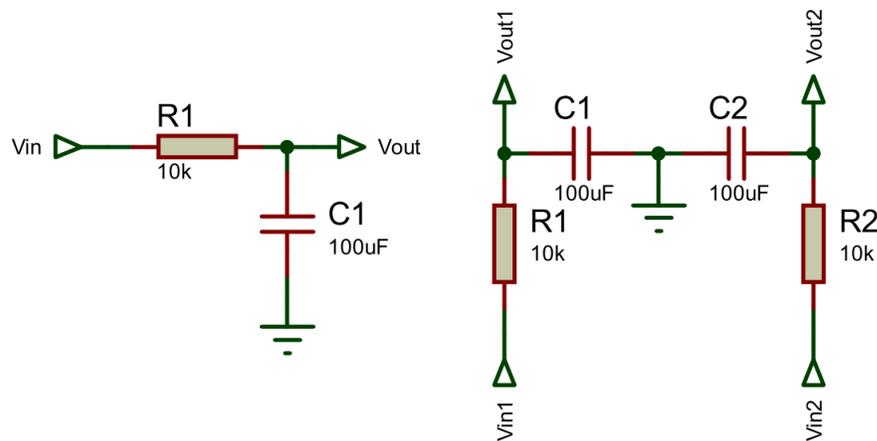


Figura 4.16: Esquema circuitual de filtros. A la izquierda se muestra el filtro sencillo y a la derecha el filtro para la señal diferencial.

Finalmente se tendrá un filtro simple para el sensor de temperatura y uno simétrico para el sensor de presión como se muestran en los diseños de las tarjetas de circuito impreso mostrados en la Figura 4.17, donde a la Izquierda se muestra el sensor de temperatura y a la derecha el de presión.

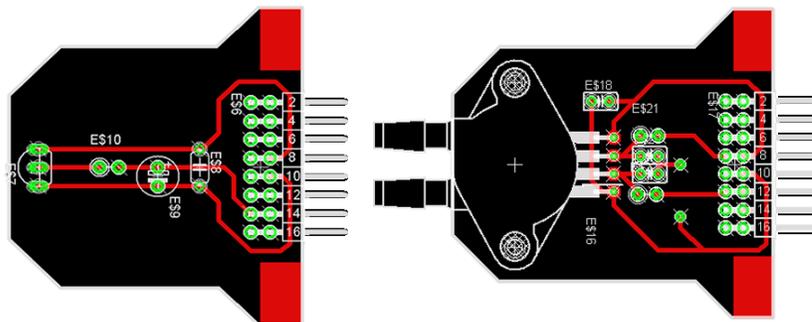


Figura 4.17: Diseño de las tarjetas de circuito impreso para los sensores. A la izquierda se muestra el sensor de temperatura y a la derecha el de presión.

#### 4.2.1.10. Diseño de Tarjeta de Pruebas

Esta tarjeta se diseñó con la finalidad de poder hacer pruebas vía USB y conectar cualquier nodo a un computador con la finalidad de poder observar procesos internos de la red como asociación, *acknowledge*, solicitudes de comandos, entre otros.

Se muestra en la Figura 4.18 el diseño de la tarjeta de circuito impreso para este nodo. Cabe destacar que el dispositivo debe estar conectado a otra fuente de alimentación para poder establecer la comunicación USB (El puerto USB no puede suministrarle energía al nodo).

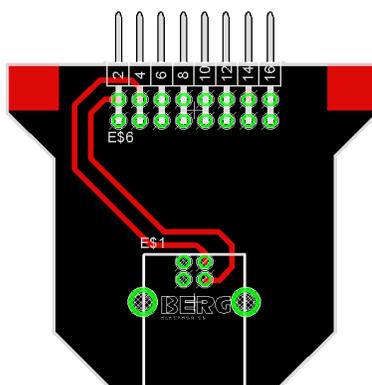


Figura 4.18: Diseño de la tarjeta de circuito impreso para la Tarjeta de Pruebas.

### 4.2.2. Fabricación de los Dispositivos

#### 4.2.2.1. Costo de los materiales

Se desglosará a continuación el costo total del proyecto en la cantidad de dispositivos desarrollados en el mismo:

**1. Nodo principal:** Se muestra en el Cuadro 4.3, el costo de un nodo principal.

Cuadro 4.3: Costos de componentes para la fabricación del Nodo Principal.

Nodo Principal

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
PIC18F27J53	Microlcontrolador	1	3,82	16,43	3,82	16,43	
MRF24J40MA	Módulo de Com. Inalámbrica	1	9,12	39,22	9,12	39,22	
NTE1904	Reg. Tensión a 3,3V	1	3,59	15,44	3,59	15,44	
Resistencias 10KΩ 1/4W	Resistencia	1	0,03	0,11	0,03	0,11	
Resistencias 1KΩ 1/4W	Resistencia	3	0,03	0,11	0,08	0,34	
Resistencias 100Ω 1/8W	Resistencia	8	0,03	0,11	0,21	0,89	
Capacitores 0,1μF 50V	Capacitor	2	0,03	0,13	0,06	0,27	
Capacitores 10μF 50V	Capacitor	1	0,03	0,13	0,03	0,13	
Puerto USB	Puerto USB A Macho	1	0,37	1,59	0,37	1,59	
Bakelita 10x15	Bakelita Doble Cara 10x15cm	1	8,62	37,07	8,62	37,07	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	25,93	111,48

**2. Nodo sensor ADC simple:** Se muestra en el Cuadro 4.4, el costo de un nodo sensor con ADC simple.

Cuadro 4.4: Costos de componentes para la fabricación del Nodo Sensor ADC Simple.

Nodo ADC Simple

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
PIC18F27J53	Microcontrolador	1	3,82	16,43	3,82	16,43	
MRF24J40MA	Módulo de Com. Inalámbrica	1	9,12	39,22	9,12	39,22	
NTE1904	Reg. 3,3V	1	3,59	15,44	3,59	15,44	
LM317	Reg. de Tensión	1	4,59	19,74	4,59	19,74	
Resistencias 10KΩ 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 1KΩ 1/4W	Resistencia	3	0,03	0,13	0,09	0,39	
Resistencias 1,1KΩ 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 470Ω 1/4W	Resistencia	2	0,03	0,13	0,06	0,26	
Resistencias 300Ω 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 100Ω 1/8W	Resistencia	8	0,03	0,13	0,24	1,03	
Resistencias 2Ω 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Capacitores 0,1μF 50V	Capacitor	2	0,03	0,13	0,06	0,26	
Capacitores 10μF 50V	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 100μF	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 220μF 16V	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 0,22μF50V	Capacitor	1	0,03	0,13	0,03	0,13	
Led Alto Brillo Blanco	Led	1	0,10	0,42	0,10	0,42	
1N4007	Diodo	3	0,07	0,30	0,21	0,90	
Zener 24V	Diodo	1	0,07	0,30	0,07	0,30	
2N3904	Transistor BJT	1	0,10	0,42	0,10	0,42	
Puerto USB A Macho	Puerto USB	1	0,04	0,16	0,04	0,16	
Bakelita Doble Cara 10x15cm	Baquelita	1	14,38	61,83	14,38	61,83	
Puente Rectificador 400V	Rectificador	1	1,71	7,36	1,71	7,36	
Jack 5mm	Jack de alimentación	1	0,89	3,83	0,89	3,83	
Conector Hembra 8x2 90 grados	Puerto de Conexiones	1	2,09	8,99	2,09	8,99	
Conector WM2S 2 PINES PCB	Conector para la Batería	1	1,01	4,34	1,01	4,34	
Portapilas 3 pilas	Portapila	1	1,26	5,42	1,26	5,42	
Interruptor 2 posiciones	Interruptor	1	2,29	9,85	2,29	9,85	
Pulsador PCB	Pulsador	1	0,05	0,22	0,05	0,22	
Pin header lineal x5	Pines de Programación	1	0,94	4,04	0,94	4,04	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	46,94	201,86

**3. Nodo sensor ADC diferencial:** Se muestra en el Cuadro 4.5, el costo de un nodo sensor con ADC diferencial.

Cuadro 4.5: Costos de componentes para la fabricación del Nodo Sensor ADC Diferencial.

Nodo ADC Diferencial

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
PIC18F27J53	Micromicrocontrolador	1	3,82	16,43	3,82	16,43	
MRF24J40MA	Módulo de Com. Inalámbrica	1	9,12	39,22	9,12	39,22	
NTE1904	Reg. 3,3V	1	3,59	15,44	3,59	15,44	
LM317	Reg. de Tensión	1	4,59	19,74	4,59	19,74	
Resistencias 10KΩ 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 1KΩ 1/4W	Resistencia	3	0,03	0,13	0,09	0,39	
Resistencias 1,1KΩ 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 470Ω 1/4W	Resistencia	2	0,03	0,13	0,06	0,26	
Resistencias 300Ω 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Resistencias 100Ω 1/8W	Resistencia	8	0,03	0,13	0,24	1,03	
Resistencias 2Ω 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Capacitores 0,1μF 50V	Capacitor	2	0,03	0,13	0,06	0,26	
Capacitores 10μF 50V	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 100μF	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 220μF 16V	Capacitor	1	0,03	0,13	0,03	0,13	
Capacitores 0,22μF50V	Capacitor	1	0,03	0,13	0,03	0,13	
Led Alto Brillo Blanco	Led	1	0,10	0,42	0,10	0,42	
1N4007	Diodo	3	0,07	0,30	0,21	0,90	
Zener 24V	Diodo	1	0,07	0,30	0,07	0,30	
2N3904	Transistor BJT	1	0,10	0,42	0,10	0,42	
Puerto USB A Macho	Puerto USB	1	0,04	0,16	0,04	0,16	
Bakelita Doble Cara 10x15cm	Baquelita	1	14,38	61,83	14,38	61,83	
Puente Rectificador 400V	Rectificador	1	1,71	7,36	1,71	7,36	
Jack 5mm	Jack de alimentación	1	0,89	3,83	0,89	3,83	
Conector Hembra 8x2 90°	Puerto de Conexiones	1	2,09	8,99	2,09	8,99	
Conector WM2S 2 PINES PCB	Conector para la Batería	1	1,01	4,34	1,01	4,34	
Portapilas 3 pilas	Portapila	1	0,15	1,26	0,15	1,26	
Interruptor 2 posiciones	Interruptor	1	2,29	9,85	2,29	9,85	
Pulsador PCB	Pulsador	1	0,05	0,22	0,05	0,22	
Pin header lineal x5	Pines de Programación	1	0,94	4,04	0,94	4,04	
MCP3204	Conv. AD de 12bits Diferencial	1	3,94	23,94	3,94	23,94	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	49,77	221,65

**4. Sensor de temperatura:** Se muestra en el Cuadro 4.6, el costo de un sensor de temperatura.

Cuadro 4.6: Costos de componentes para la fabricación del sensor de temperatura.

Sensor temperatura

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
Conector pin header 8x2 90°	Puerto de conexión	1	2,09	8,99	2,09	8,99	
Capacitores 0,1μF 50V	Capacitor	1	0,03	0,13	0,03	0,13	
Resistencias 10KΩ 1/4W	Resistencia	1	0,03	0,13	0,03	0,13	
Bakelita Doble Cara 5x5cm	Baquelita	1	9,30	39,99	9,30	39,99	
Capacitores 100μF	Capacitor	1	0,03	0,13	0,03	0,13	
MCP9700A	Sensor de temperatura	1	6,00	25,80	6,00	25,80	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	17,48	75,164

**5. Sensor de presión:** Se muestra en el Cuadro 4.7, el costo de un sensor de presión.

Cuadro 4.7: Costos de componentes para la fabricación del sensor de presión.

Sensor Presión

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
Conector pin header 8x2 90°	Puerto de conexión	1	2,09	8,99	2,09	8,99	
Capacitores 0,1µF 50V	Capacitor	1	0,03	0,13	0,03	0,13	
Resistencias 10KΩ 1/4W	Resistencia	2	0,03	0,13	0,06	0,26	
Bakelita Doble Cara 5x5cm	Baquelita	1	9,30	39,99	9,30	39,99	
Capacitores 100µF	Capacitor	2	0,03	0,13	0,06	0,26	
MPX53DP	Sensor de presión diferencial	1	8,69	37,37	8,69	37,37	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	20,23	86,989

6. **Tarjeta de pruebas:** Se muestra en el Cuadro 4.8, el costo de la tarjeta de pruebas.

Cuadro 4.8: Costos de componentes para la fabricación de la tarjeta de pruebas.

Tarjeta Pruebas

Componente	Descripción	Cantidad	USD/Unidad	BsF/Unidad	Total USD	Total BsF	
Conector pin header 8x2 90°	Puerto de conexión	1	2,09	8,99	2,09	8,99	
Puerto USB tipo B Hembra	Puerto USB	1	3,98	17,11	3,98	17,11	
Bakelita Doble Cara 5x5cm	Baquelita	1	8,62	37,07	8,62	37,07	
Factor de control cambiario 1USD = 4,3BsF. (2013)					Total=	14,69	63,167

7. **Costo total del proyecto:** Se muestra en el Cuadro 4.9, el costo total del proyecto.

Cuadro 4.9: Costo total del proyecto.

Equipos	Costo
Nodo Principal	111,4775
Nodo Sensor ADC Simple (x2)	403,72614
Nodo Sensor ADC Diferencial	221,64507
Sensor Temperatura (x2)	150,328
Sensor Presión	86,989
Tarjeta de Pruebas	63,167
Gastos de Envío	165,59991
Total=	1202,9326

#### 4.2.2.2. Dispositivos Terminados

1. **Nodo principal:** Se muestra en la Figura 4.19, el dispositivo correspondiente al nodo principal construido.

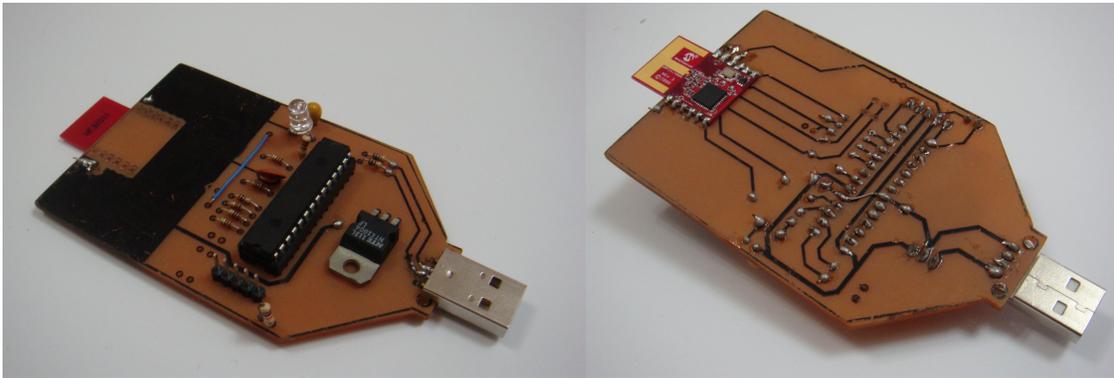


Figura 4.19: Nodo principal fabricado.

**2. Nodo sensor ADC simple:** Se muestra en la Figura 4.19, el dispositivo correspondiente al nodo sensor ADC simple construido.

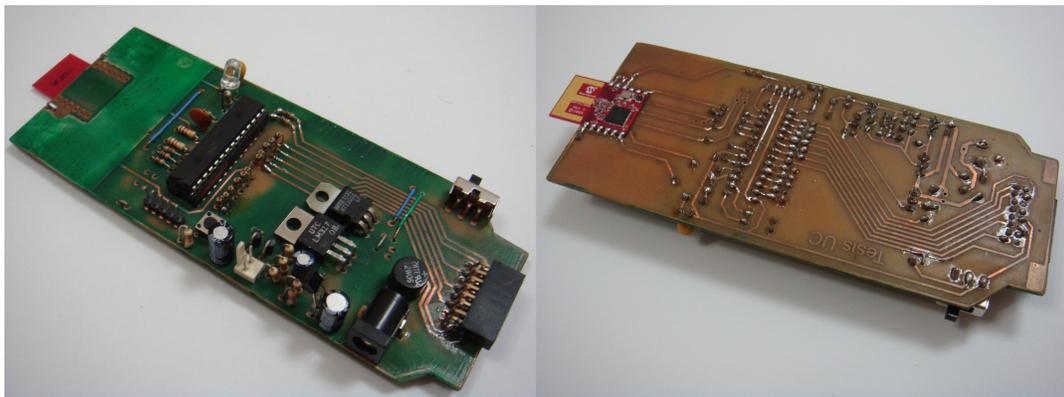


Figura 4.20: Nodo sensor con ADC simple fabricado.

**3. El nodo sensor ADC diferencial:** Se muestra en la Figura 4.19, el dispositivo correspondiente al nodo sensor ADC diferencial construido.

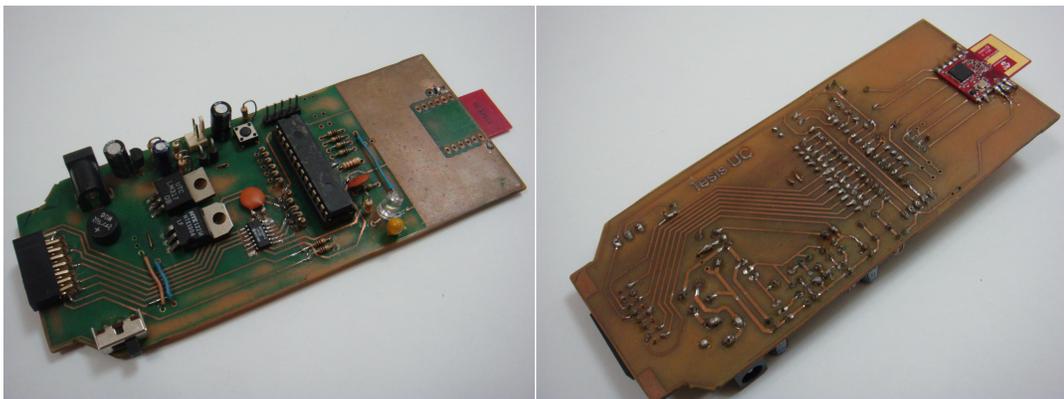


Figura 4.21: Nodo sensor con ADC diferencial fabricado.

**4. Sensor de temperatura:** Se muestra en la Figura 4.19, el dispositivo correspondiente al sensor de temperatura construido.

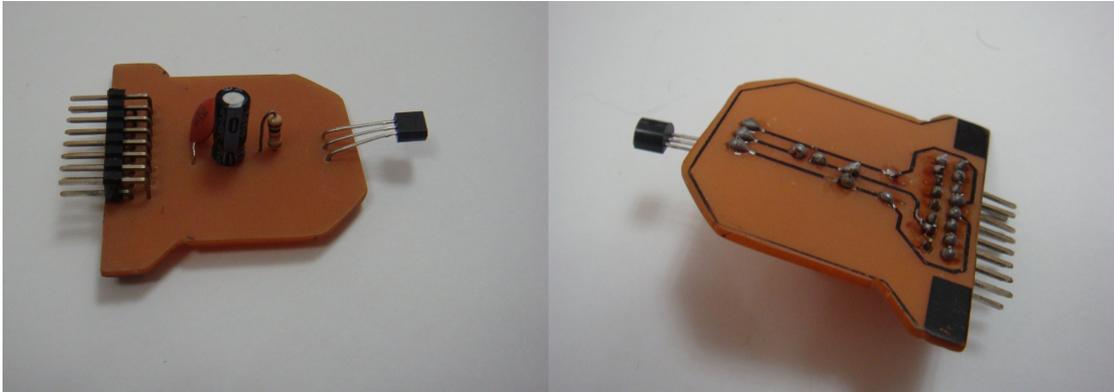


Figura 4.22: Sensor de Temperatura fabricado.

**5. Sensor de presión:** Se muestra en la Figura 4.19, el dispositivo correspondiente al sensor de presión construido.

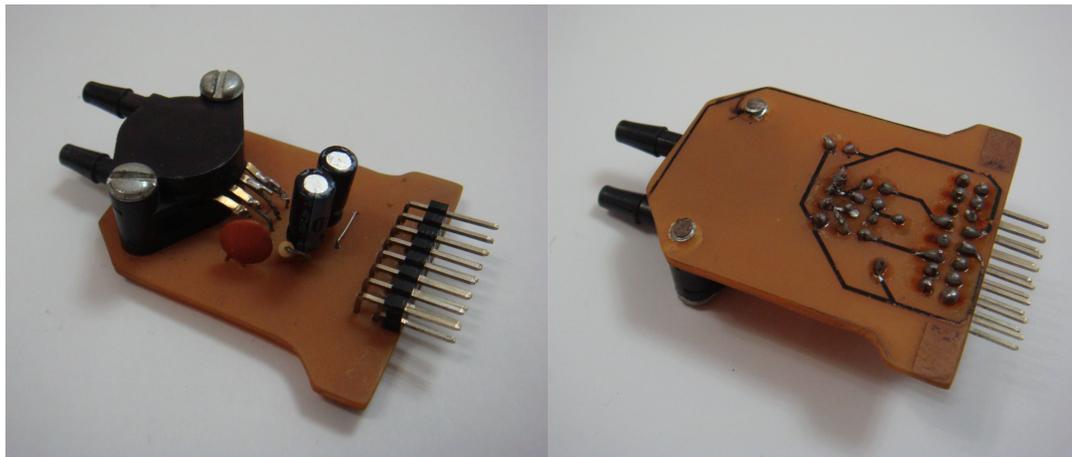


Figura 4.23: Sensor de Presión fabricado.

### 4.3. El Protocolo de la Capa de Aplicación

Esta sección tiene como objetivo el desarrollo de un algoritmo de

Para la comunicación entre el computador y un dispositivo conectado USB se diseñó un protocolo de comunicación de fácil manejo y edición, en caso de ser necesario, a través de un factor de ajuste llamado *command factor*, factor ajusta los 4 bits más significativos del comando de modo de poder desplazarlos todos en caso de que algún comando sea necesario para alguna aplicación externa.

### 4.3.1. Estructura básica

Los comandos han de ser enviados vía USB siguiendo la estructura mostrada en la Figura 4.24.

Ocetos:	1	1	1	Variable
Nombre:	SrcAddH	SrcAddL	Comando	Carga
	Identificador de Red		Identificador de Comando	Bytes a transferir

Figura 4.24: Estructura básica del protocolo

En donde se tiene que el identificador de red deben ser dos octetos que contengan la dirección corta de 16 bits del dispositivo a quien se le quiera enviar el comando, seguidamente un octeto cuyo contenido debe ser el comando deseado y finalmente la carga de longitud variable que puede ser de 0 a 14 octetos según el comando escogido.

### 4.3.2. Comandos

En el Cuadro 4.10, se puede observar la lista de comandos establecidos con su respectiva descripción y la cantidad de octetos que debe contener la carga y la cantidad de octetos que se han de esperar como carga de la respuesta, si el comando devuelve alguna.

Cuadro 4.10: Cuadro de comandos del protocolo de comunicación para la capa de aplicación.

Cmd	Descripción	Carga	
		Solicitud	Respuesta
11	Enviar un reset	0	-
12	Solicitar último paquete enviado	0	0-128
13	Solicitar el último paquete recibido	0	0-128
14	Solicitar lista de nodos cercanos	0	x12
15	Solicitar lista de rutas	0	x6
16	Solicitar lista de descubrimiento de rutas	0	x8
17	Solicitar lista de Errores	0	x12
18	Solicitar lista de canales ocupados	0	x3
19	Solicitar medición ADC	0	15
1A	Solicitar rango de direcciones para asignación	0	4
1B	Lectura de un registro del MRF de corto direccionamiento	1	1
1C	Lectura de un registro del MRF de largo direccionamiento	2	1
1D	Escritura de un registro del MRF de corto direccionamiento	2	-
1E	Escritura de un registro del MRF de largo direccionamiento	3	-
1F	Solicitar el canal de operación a un nodo	0	1
20	Fijar el canal de operación de un nodo	1	-
21	Solicitud de la potencia de la antena	0	

Cuadro 4.11: Continuación de la tabla de comandos del protocolo de comunicación para la capa de aplicación.

Cmd	Descripción	Carga	
		Solicitud	Respuesta
22	Fijar la potencia de la antena	1	-
23	Solicitar el tipo de sensor	0	1
24	Solicitar el estado de la batería	0	15
25	Solicitar unidad asociada a la medición	0	1
26	Fijar la unidad de medición	1	-
27	Solicitar el offset	0	2
28	Fijar el offset	2	-
29	Solicitar el valor de escala	0	2
2A	Fijar el factor de escala	2	-
2B	Solicitar el valor del Filtro	0	1
2C	Fijar el Valor del Filtro	1	-
2D	Solicitar la Cantidad de muestras	0	1
2E	Fijar la cantidad de muestras	1	-
2F	Solicitar Registro de Calibración 1	0	2
30	Fijar Registro de Calibración 1	2	-
31	Solicitar Registro de calibración 2	0	2
32	Fijar Registro de calibración 2	2	-
33	Solicitar Registro de Información 1	0	12
34	Fijar Registro de Información 1	12	-
35	Solicitar Registro de Información 2	0	12
36	Fijar Registro de Información 2	12	-
37	Solicitar Registro de Información 3	0	12
38	Fijar Registro de Información 3	12	-
39	Solicitar Registro de Información 4	0	12
3A	Fijar Registro de Información 4	12	-
3B	Solicitar Registro de Información 5	0	12
3C	Fijar Registro de Información 5	12	-
3D	Solicitar Registro de Información 6	0	12
3E	Fijar Registro de Información 6	12	-
3F	Reservado	-	-
40	Enviar un reset a toda la red	0	-
41	Fijar el canal de operación de la red	1	-
42	Solicitar la tasa de de muestreo del conversor ADC	0	2
43	Fijar tasa de muestreo	2	-
44	Respalda información del coordinador en su memoria flash	0	-
45	Ajustar la Fecha y Hora	7	-

#### 4.3.2.1. Enviar un *reset* (Comando 0x11)

Para reiniciar un dispositivo específico, se debe enviar el comando 11 a la dirección corta de 16 bits del dispositivo que se desea reiniciar, este comando no genera ninguna respuesta por parte del dispositivo.

Octetos: 1	1	1
SrcAddH	SrcAddL	0x11

Figura 4.25: Envío de *reset* a un dispositivo

#### 4.3.2.2. Enviar un *reset* a toda la red (Comando 0x40)

Para reiniciar la red se debe enviar el comando 40 como un broadcast, es decir, con la dirección corta de 16 bits en 0xFFFF, este comando no genera ninguna respuesta.

Octetos: 1	1	1
0xFF	0xFF	0x40

Figura 4.26: Envío de *reset* a la red

#### 4.3.2.3. Respalda información del coordinador en su memoria flash (Comando 0x44)

Para respaldar toda la información pertinente a la red en la memoria flash interna del dispositivo se debe enviar el comando 44 a la dirección del dispositivo al cual se le desee respaldar esta información. Este comando solo es válido para el coordinador de la PAN.

Octetos: 1	1	1
0x00	0x00	0x44

Figura 4.27: Respalda información del coordinador

#### 4.3.2.4. Solicitar información

Para solicitar cualquier información a un dispositivo cualquiera se debe enviar el comando correspondiente a la información que se desea obtener a la dirección corta de 16 bits del dispositivo en cuestión.

**Solicitar último paquete enviado o recibido (Comando 0x12, Comando 0x13)** Al solicitar esta información, enviando el comando 12 o 13 respectivamente, se recibirá un paquete de longitud variable que contendrá la siguiente información: dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando correspondiente a la solicitud realizada y finalmente el último paquete transmitido o recibido inalámbricamente por el dispositivo en cuestión.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x12	

Octetos: 1	1	1	0-108
SrcAddH	SrcAddL	0x12	Dato enviado

Figura 4.28: Estructura de solicitud y respuesta del ultimo paquete enviado

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x13	

Octetos: 1	1	1	0-108
SrcAddH	SrcAddL	0x13	Dato recibido

Figura 4.29: Estructura de solicitud y respuesta del ultimo paquete recibido

**Solicitar lista de nodos cercanos (Comando 0x14)** La solicitud de la lista de nodos cercanos, realizada al enviar el comando 14, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizo la solicitud, el comando de la solicitud realizada, un octeto correspondiente a la cantidad de dispositivos en dicha lista y doce octetos correspondientes a la información de red de un dispositivo de dicha lista, esta respuesta se recibe hasta que se hayan transmitido todos los dispositivos en la lista.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x14	

Octetos: 1	1	1	13
SrcAddH	SrcAddL	0x14	Nodo cercano

Figura 4.30: Estructura de solicitud y respuesta de la lista de nodos cercanos

**Solicitar lista de rutas (Comando 0x15)** Al solicitar la lista de rutas, realizada al enviar el comando 15, genera como respuesta la dirección corta dos octetos con de 16 bits correspondiente al dispositivo al que se le realizo la solicitud, el comando de la solicitud realizada, un octeto correspondiente a la cantidad de dispositivos en dicha lista y seis octetos correspondientes a la información de red de un dispositivo de dicha lista, esta respuesta se recibe hasta que se hayan transmitido todos los dispositivos en la lista.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x15	

Octetos: 1	1	1	6
SrcAddH	SrcAddL	0x15	Ruta

Figura 4.31: Estructura de solicitud y respuesta de la lista de rutas

**Solicitar lista de descubrimiento de rutas (Comando 0x16)** Cuando se solicita la lista de descubrimiento de rutas, realizada al enviar el comando 16, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y ocho octetos correspondientes a la información de red de un dispositivo de dicha lista, esta respuesta se recibe hasta que se hayan transmitido todos los dispositivos en la lista.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x16	

Octetos: 1	1	1	8
SrcAddH	SrcAddL	0x16	Posible ruta

Figura 4.32: Estructura de solicitud y respuesta de la lista de posibles rutas

**Solicitar lista de errores (Comando 0x17)** La solicitud de la lista de errores, realizada al enviar el comando 17, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y doce octetos correspondientes a la información del error y la fecha del suceso, esta respuesta se recibe hasta que se hayan transmitido todos los errores en la lista.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x17	

Octetos: 1	1	1	12
SrcAddH	SrcAddL	0x17	Error

Figura 4.33: Estructura de solicitud y respuesta de la lista de errores

**Solicitar lista de canales ocupados (Comando 0x18)** Esta solicitud, realizada al enviar el comando 18, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y tres octetos correspondientes a la información de los canales ocupados registrados

por el dispositivo, esta respuesta se recibe hasta que se hayan transmitido todos los canales ocupados en la lista.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x18	

Octetos: 1	1	1	3
SrcAddH	SrcAddL	0x18	Canal ocupado

Figura 4.34: Estructura de solicitud y respuesta de la lista de canales ocupados

**Solicitar medición ADC (Comando 0x19)** Esta solicitud, realizada al enviar el comando 19, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y tres octetos, donde los primeros dos corresponden a la medición y el último a su signo.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x19	

Octetos: 1	1	1	3
SrcAddH	SrcAddL	0x19	Medición

Figura 4.35: Estructura de solicitud y respuesta de la medición

**Solicitar rango de direcciones para asignación (Comando 0x1A)** La solicitud del rango de direcciones cortas de 16 bits permitidas para asignar a nuevos dispositivos, realizada al enviar el comando 1A, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y cuatro octetos siendo los primeros dos octetos el menor valor que puede tener las direcciones permitidas para asignar y los últimos dos octetos el máximo valor de la misma.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x1A	

Octetos: 1	1	1	13
SrcAddH	SrcAddL	0x1A	Direcciones para asignar

Figura 4.36: Estructura de solicitud y respuesta del rango de direcciones disponible a asignar

**Lectura de un registro del MRF de corto direccionamiento (Comando 0x1B)** Al solicitar el valor en un registro corto del módulo de comunicación inalámbrica, realizada al

enviar el comando 1B y como carga un octeto con el registro a ser solicitado, se genera como respuesta un octeto con el valor en dicho registro.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x1B	Registro corto

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x1B	Valor

Figura 4.37: Estructura de solicitud y respuesta de un registro de corto direccionamiento del módulo MRF24j40

**Lectura de un registro del MRF de largo direccionamiento (Comando 0x1C)** Al solicitar el valor en un registro largo del modulo de comunicación inalámbrica, realizada al enviar el comando 1C y como carga dos octeto con el registro a ser solicitado, se genera como respuesta un octeto con el valor en dicho registro.

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x1C	Registro largo

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x1C	Valor

Figura 4.38: Estructura de solicitud y respuesta de un registro de largo direccionamiento del módulo MRF24j40

**Solicitar canal de operación a un nodo (Comando 0x1F)** Al solicitar el canal de operación, realizada al enviar el comando 1F, se genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizo la solicitud, el comando de la solicitud realizada y un octeto que indica el canal en el cual se encuentra operando el dispositivo en los 4 bits mas significativos del mismo.

Octetos: 1	1	1
SrcAddH	SrcAddL	0x1F

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x1F	Canal

Figura 4.39: Estructura de solicitud y respuesta del canal de operación

**Solicitud de la potencia de la antena (Comando 0x21)** Esta solicitud, realizada al enviar el comando 21, genera como respuesta dos octetos con la dirección corta de 16 bits

correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica la potencia de transmisión del dispositivo, de la forma mostrada en la figura 4.41.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x21	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x21	Potencia

Figura 4.40: Estructura de solicitud y respuesta de la potencia de transmisión

bit 7-6	<b>Large Scale Control for TX Power bits</b> 11 = -30 dB 10 = -20 dB 01 = -10 dB 00 = 0 dB
bit 5-3	<b>Small Scale Control for TX Power bits</b> 111 = -6.3 dB 110 = -4.9 dB 101 = -3.7 dB 100 = -2.8 dB 011 = -1.9 dB 010 = -1.2 dB 001 = -0.5 dB 000 = 0 dB
bit 2-0	<b>Reserved:</b> Maintain as '0'

Figura 4.41: Niveles ajustables de potencia de transmisión del MRF24J40MA.

**Solicitar el tipo de sensor (Comando 0x23)** La solicitud del tipo de sensor, realizada al enviar el comando 23, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica que tipo de sensor debe ser conectado al dispositivo.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x23	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x23	Tipo de sensor

Figura 4.42: Estructura de solicitud y respuesta del tipo de sensor

**Solicitar estado de la batería (Comando 0x24)** Al solicitar el estado de la batería del dispositivo, realizada al enviar el comando 24, se genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica el porcentaje restante de la batería.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x24	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x24	Estado de la batería

Figura 4.43: Estructura de solicitud y respuesta del estado de la batería

**Solicitar unidad asociada a la medición (Comando 0x25)** Esta solicitud, realizada al enviar el comando 25, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica la unidad en la que debe ser mostrada la medición, siguiendo la información mostrada en la tabla 4.12.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x25	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x25	Unidad de medición

Figura 4.44: Estructura de solicitud y respuesta de la unidad de medición

Cuadro 4.12: Unidades de medición

Valor	Unidad
1	°C
2	°F
3	psi
4	bar

**Solicitar offset o escala de la medición (Comando 0x27, Comando 0x29)** Al solicitar esta información, enviando el comando 27 o 29 respectivamente, se recibirá un paquete que contendrá la siguiente información: dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando correspondiente a la solicitud realizada y finalmente dos octetos que contendrán el valor con signo de la información solicitada.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x27	

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x27	Offset

Figura 4.45: Estructura de solicitud y respuesta del offset de la medición

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x29	

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x29	Escala

Figura 4.46: Estructura de solicitud y respuesta de la escala de la medición

**Solicitar tipo de filtro (Comando 0x2B)** La solicitud del tipo de filtro digital, realizada al enviar el comando 2B, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica que tipo de filtro digital que se le ha de aplicar a la medición.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x2B	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x2B	Filtro digital

Figura 4.47: Estructura de solicitud y respuesta del filtro digital a usar

**Solicitar la cantidad de muestras (Comando 0x2D)** Esta solicitud, realizada al enviar el comando 2D, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y un octeto que indica la cantidad de mediciones que serán promediadas antes del mostrar el resultado final.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x2D	

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x2D	Cantidad de mediciones promediadas

Figura 4.48: Estructura de solicitud y respuesta de la cantidad de mediciones que serán promediadas

**Solicitar registros de calibración (Comando 0x2F, Comando 0x31)** Al solicitar esta información, enviando el comando 2F para el registro de calibración 1 o 31 para el registro de calibración 2, se recibirá un paquete que contendrá la siguiente información: dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando correspondiente a la solicitud realizada y finalmente dos octetos que contendrán el valor sin signo de la información solicitada.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x2F ó 0x31	

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x2F ó 0x31	Calibración

Figura 4.49: Estructura de solicitud y respuesta de los registros de calibración

**Solicitar tipo de registros de información (Comando 0x33, 0x35, 0x37, 0x39, 0x3B, 0x3D)** La solicitud de cualquier registro de información, realizada al enviar cualquiera de los comandos 33, 35, 37, 39, 3B o 3D, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y doce octetos que contienen una cadena de caracteres que representan el texto contenido en el registro de información solicitado.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x33, 0x35, 0x37, 0x39, 0x3B ó 0x3D	

Octetos: 1	1	1	12
SrcAddH	SrcAddL	0x33, 0x35, 0x37, 0x39, 0x3B ó 0x3D	Información

Figura 4.50: Estructura de solicitud y respuesta de los registros de información

**Solicitar la tasa de muestreo (Comando 0x42)** Esta solicitud, realizada al enviar el comando 42, genera como respuesta dos octetos con la dirección corta de 16 bits correspondiente al dispositivo al que se le realizó la solicitud, el comando de la solicitud realizada y dos octetos que indica el tiempo entre las mediciones como un múltiplo de 20ms.

Octetos: 1	1	1	
SrcAddH	SrcAddL	0x42	

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x42	Intervalo

Figura 4.51: Estructura de solicitud y respuesta del intervalo entre muestras

#### 4.3.2.5. Guardado de datos

Los comandos que fijan información no generan ninguna respuesta. Estos comandos se envían a la dirección corta de 16 bits del dispositivo al cual se le desee fijar una información nueva, seguido por el comando respectivo y finalmente, como carga, la información nueva a guardar.

**Escritura de un registro del MRF de direccionamiento corto (Comando 0x1D)** Para fijar un valor en uno de los registros cortos del modulo de comunicación inalámbrica la carga de ser de dos octetos que contengan el registro a ser modificado en un octeto y el valor que va a ser asignado en un octeto.

Octetos: 1	1	1	1	1
SrcAddH	SrcAddL	0x1D	Registro corto	Valor

Figura 4.52: Estructura de la escritura de un registro de direccionamiento corto

**Escritura de un registro del MRF de direccionamiento corto (Comando 0x1E)** Para fijar un valor en uno de los registros largos del modulo de comunicación inalámbrica la carga de ser de dos octetos que contengan el registro a ser modificado en dos octetos y el valor que va a ser asignado en un octeto.

Octetos: 1	1	1	2	1
SrcAddH	SrcAddL	0x1E	Registro Largo	Valor

Figura 4.53: Estructura de la escritura de un registro de direccionamiento largo

**Fijar canal de operación de un nodo (Comando 0x20)** Para fijar el canal de operación la carga debe ser de un octeto, de tal forma que el canal de operación se encuentre el los cuatro bits mas significativos y los 4 bits menos significativos estén fijos en 3.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x20	Canal

Figura 4.54: Estructura de la escritura del canal

**Fijar potencia de la antena (Comando 0x22)** Para fijar la potencia de transmisión la carga debe ser de un octeto, de tal forma que siga la información en la figura 4.41.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x22	Potencia

Figura 4.55: Estructura del cambio de la potencia de transmisión

**Fijar unidad de medición (Comando 0x26)** Para fijar la unidad de medición la carga debe ser de un octeto, de la forma mostrada de la tabla 4.12.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x26	Unidad

Figura 4.56: Estructura del cambio de la unidad de medición

**Fijar offset o escala de medición (Comando 0x28, Comando 0x2A)** Para fijar el offset o la escala de la medición la carga debe ser de 2 octetos, de modo que sean un número entero comprendido entre -32767 y 32768, siendo el signo el bit mas significativo.

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x28	Offset

Figura 4.57: Estructura del cambio del offset

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x2A	Escala

Figura 4.58: Estructura del cambio de la escala

**Fijar filtro digital (Comando 0x2C)** Para fijar el tipo de filtro digital aplicado a la medición, la carga debe ser de 1 octeto.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x2C	Filtro

Figura 4.59: Estructura del cambio del filtro

**Fijar la cantidad de mediciones a ser promediadas (Comando 0x2E)** Para fijar la cantidad de mediciones a ser promediadas la carga debe estar expresada en un 1 octeto.

Octetos: 1	1	1	1
SrcAddH	SrcAddL	0x2E	Cantidad de mediciones promediadas

Figura 4.60: Estructura del cambio de la cantidad de mediciones a ser promediadas

**Fijar registros de calibración (Comando 0x30, Comando 0x32)** Para fijar los registros de calibración la carga de ser dos octetos que contendrán el valor sin signo a asignar a estos registros.

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x30 ó 0x32	Calibracion

Figura 4.61: Estructura del cambio de los registros de calibración

**Fijar registros de información (Comando 0x33, 0x35, 0x37, 0x39, 0x3B, 0x3D)** Para fijar los registros de información la carga debe ser de doce octetos que contengan una cadena de caracteres.

Octetos: 1	1	1	12
SrcAddH	SrcAddL	0x33, 0x35, 0x37, 0x39, 0x3B ó 0x3D	Información

Figura 4.62: Estructura del cambio de los registros de información

**Fijar canal de operación de la red (Comando 0x41)** Para fijar el canal de operación a toda la red se debe enviar el paquete como un broadcast, esto significa enviar el paquete con destino 0xFFFF, y la carga debe ser de un octeto, de tal forma que el canal de operación se encuentre en los cuatro bits más significativos y los 4 bits menos significativos estén fijos en 3.

Octetos: 1	1	1	1
0xFF	0xFF	0x41	Canal

Figura 4.63: Estructura del cambio del canal de la red

**Fijar intervalo de tiempo entre muestras (Comando 0x43)** Para fijar el intervalo de tiempo entre muestras la carga debe ser de dos octetos que indica el tiempo entre las mediciones como un múltiplo de 20ms.

Octetos: 1	1	1	2
SrcAddH	SrcAddL	0x43	Intervalo

Figura 4.64: Estructura del cambio de la cantidad de mediciones a ser promediadas

**Fijar fecha y hora (Comando 0x45)** Para fijar la fecha y la hora a la red se debe enviar como carga siete octetos que contenga la información en BCD como se muestra en la tabla 4.65.

Octetos: 1	1	1	1	1	1	1	1	1	1
SrcAddH	SrcAddL	0x45	Día	Mes	Año	Día de la semana	Hora	Minutos	Segundos

Figura 4.65: Estructura del cambio de la fecha

## 4.4. Desarrollo del Algoritmo

Esta sección tiene como objetivo el desarrollo del algoritmo que le brinda a cada nodo la capacidad de detección, comunicación y creación de rutas entre los nodos de la red además de los procesos de asociación a la red, medición de baterías, listado de errores entre otros. Dichos algoritmos son descritos a continuación:

### 4.4.1. Algoritmo de Inicio

En todo código debe haber un inicio de programa, en este caso para inicialización de variables a utilizar, configuración de todos los registros pertinentes del microcontrolador para lograr el funcionamiento deseado, entre otros. Se muestra en la Figura 4.66, el diagrama de bloques generalizado, correspondiente al algoritmo de inicio del código del microcontrolador según lo descrito previamente.

### 4.4.2. Algoritmo de Interrupciones

El código del microcontrolador está diseñado para manejar la gran mayoría de sus procesos mediante la Rutina de Servicio de Interrupción. Esta posee cuatro fuentes que pueden invocarla: USB, WDT, MRF, Timer1, como se muestra en la Figura 4.67.

Las interrupciones por USB están relacionadas con el protocolo de la capa de aplicación; el factor variable se encuentra en el destinatario del comando recibido. Por lo tanto, como puede observarse en la Figura 4.67, seguido de una interrupción USB es necesario discernir

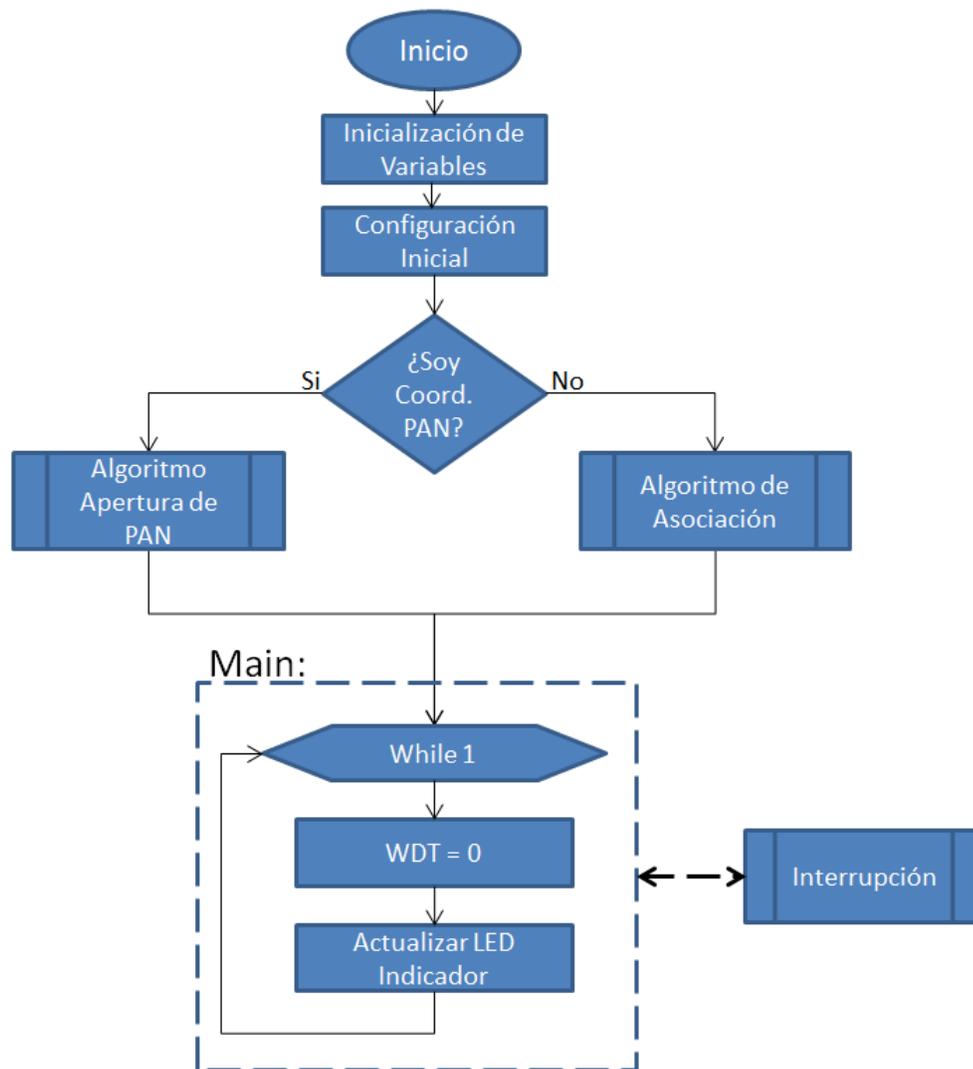


Figura 4.66: Diagrama de bloques del algoritmo de inicio del código del microcontrolador.

si el nodo principal (conectado al computador), es el destino del comando enviado vía USB<sup>1</sup>; de ser así este debe procesar la información internamente en función a los comandos del protocolo de la capa de aplicación; por otro lado, si el destinatario es otro nodo en la red el microcontrolador debe cargar la información bajo una trama ya sea de tipo comando o tipo datos (según sea el caso), y llamar a la rutina de solicitud de ruta para su envío.

Las interrupción por *watch dog timer* (WDT) indica que el microcontrolador no recibió un paquete inalámbrico con su *short address* en el tiempo preestablecido, indicando un mal funcionamiento del dispositivo o que ha quedado huérfano por lo que se induce un *reset* en el microcontrolador y el módulo de comunicación inalámbrica.

Las interrupciones producidas por el Módulo de Radio Frecuencia (MRF), pueden ser

<sup>1</sup>La comunicación USB requiere una interrupción periódica en intervalos de entre 1ms y 10ms. Esta interrupción no está considerada dentro del esquema de interrupción.

producto o de la recepción o la correcta transmisión de un paquete. En el caso de la recepción se verifica, si el destino del paquete coincide con el *short address* del dispositivo, de no serlo el paquete se retransmite; en caso de ser así, se verifica que el paquete recibido forme parte o no del proceso de asociación. De formar parte, el paquete puede llegar a diferentes puntos dentro del proceso de asociación en función al tipo de paquete recibido como se mostrará más adelante dentro de este trabajo de grado; De no formar parte indica que el paquete recibido debe ser procesado bajo el protocolo de la capa de aplicación.

Finalmente las interrupciones por *Timer1* indican que debe tomarse una muestra del conversor analógico digital.

Todos los procesos descritos pueden observarse en la Figura 4.67.

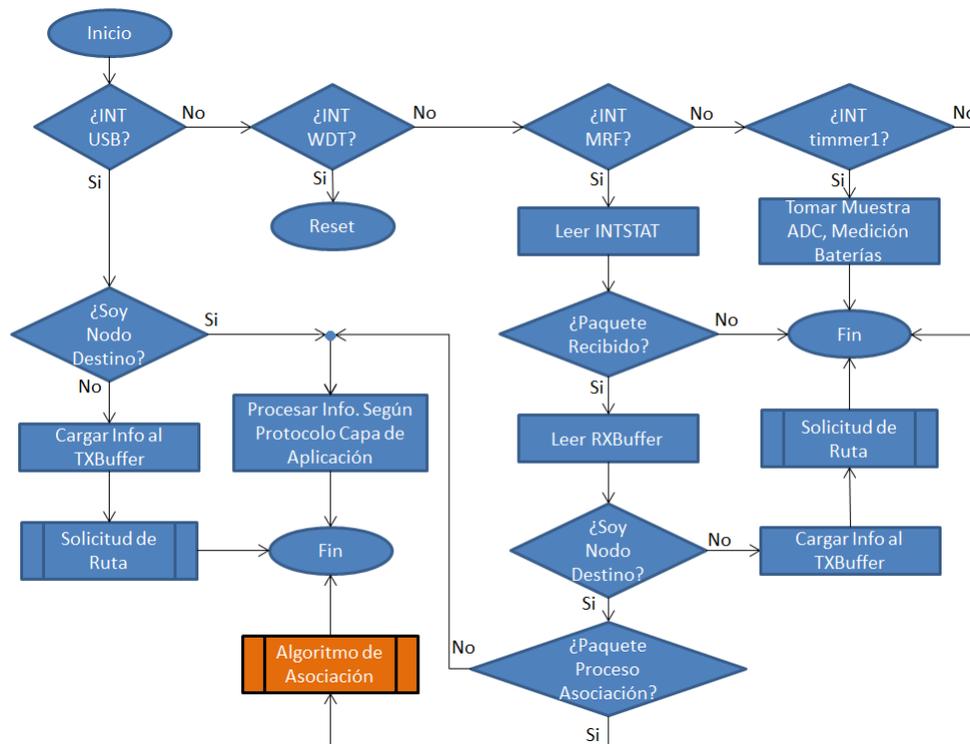


Figura 4.67: Diagrama de bloques del algoritmo de interrupciones.

### 4.4.3. Algoritmo de Apertura de PAN

Una WPAN, requiere ser aperturada por su coordinador. Esto implica configurar el módulo de comunicación inalámbrica, fijar la configuración del entramado, fijar modos de operación, conseguir un canal libre (o con el menor uso posible), entre otros; y finalmente entrar en el estado de aceptación de nuevos nodos a la red ya que una red de un solo dispositivo no es de gran utilidad. Se muestra, en la Figura 4.68, el proceso de apertura de la red por parte del Coordinador de PAN. Cabe destacar que esta rutina posee un mecanismo, en dado caso de estar todos los canales ocupados, buscar el de menor tráfico y establecerse en él.

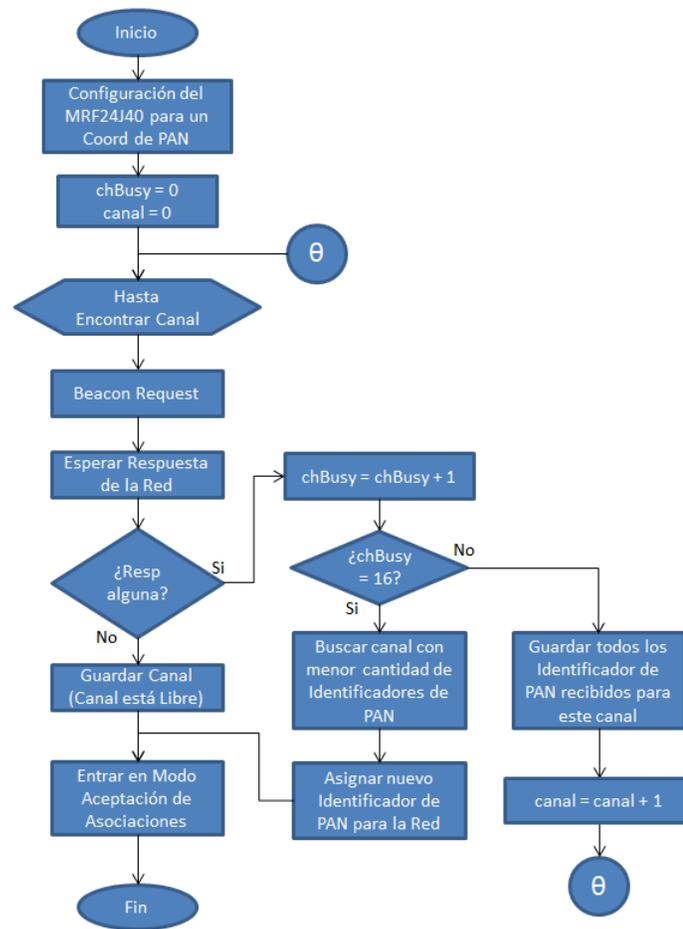


Figura 4.68: Diagrama de bloques del algoritmo de apertura de red.

#### 4.4.4. Algoritmo de Asociación

Como fue mencionado anteriormente, una red conformada por un solo dispositivo no es de utilidad, motivo por el cual es imprescindible que toda WPAN posea un proceso de asociación que la permita añadir nuevos elementos a esta. En el caso de la red planteada en este trabajo especial grado, este proceso pasa a ser de mucha mayor importancia debido al corto alcance de los módulos de comunicación inalámbrica y la necesidad de, mediante el uso de la topología Mesh y las bondades de enrutamiento de la capa de red (ZigBee), brindarle a esta red un área de cobertura mucho mayor ( $\text{Radius} > 1$ ).

En este proceso se inicia, primeramente con la configuración del dispositivo como nodo y se ajusta el canal en cero para iniciar la búsqueda de la WPAN para asociación. Luego, en la Figura 4.69, se observa el proceso descrito en la sección 2.2.2.6, donde se inicia el escaneo de los canales haciendo uso de un *beacon request* para detectar si existe una red ocupando el canal, de no estarlo es necesario el cambio de canal y reenviar nuevamente la solicitud; si por el contrario es recibido un *beacon response*, se envía una petición de asociación (Association Request), si esta no es respondida se envía una Confirmación de Asociación (Association Confirm); si por el contrario se recibe la respuesta de asociación (Association Response)

, se realiza, en un ciclo, la Petición de Rango si esta es respondida finaliza el proceso de asociación de lo contrario se realizarán intentos de petición hasta que este sea enviado o hasta que caduque el tiempo del WDT y se produzca un *reset*, reiniciando todo el proceso.

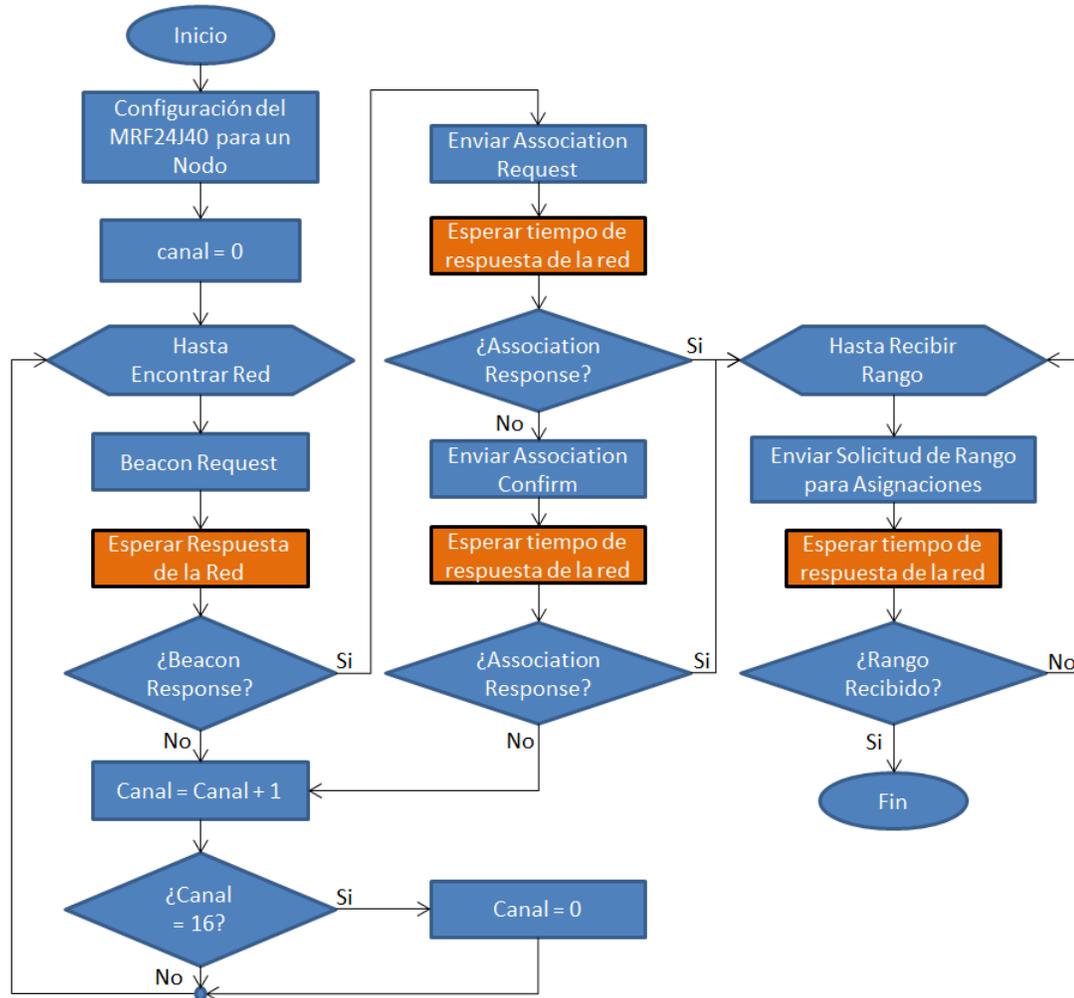


Figura 4.69: Diagrama de bloques del algoritmo de asociación a la red. Se marcan de color naranja los diferentes puntos de acceso al algoritmo, dependiendo del paquete perteneciente al proceso de asociación que se reciba.

#### 4.4.5. Algoritmo Enviar Paquete

En el caso que sea necesario enviar un paquete existen tres posibilidades, que el nodo destino esté dentro del alcance del dispositivo que desea hacer el envío, que este fuera de este pero esté en la lista de rutas y que este fuera de alcance y tampoco aparezca en lista de rutas. En caso de que el dispositivo se encuentre dentro del alcance este es enviado directamente sin necesidad de realizar ningún tipo de proceso de enrutamiento; En caso de que esté listado en la tabla de rutas se arma el paquete con identificador de destino el primer nodo de la ruta en dirección al nodo destino; y como última posibilidad, en el caso de que no se encuentre listado como dispositivo dentro del rango de alcance o como una ruta previamente establecida, se

procede a iniciar el proceso de enrutamiento, el cual comienza enviando una solicitud de rutas a todos los dispositivos al alcance (broadcast) y se espera a que el dispositivo destino confirme una ruta para poder enviar el paquete o a que los dispositivos cercanos indiquen que no es posible comunicarse con ese destino en cuyo caso se lista el error. Este proceso se puede observar en la Figura 4.70.

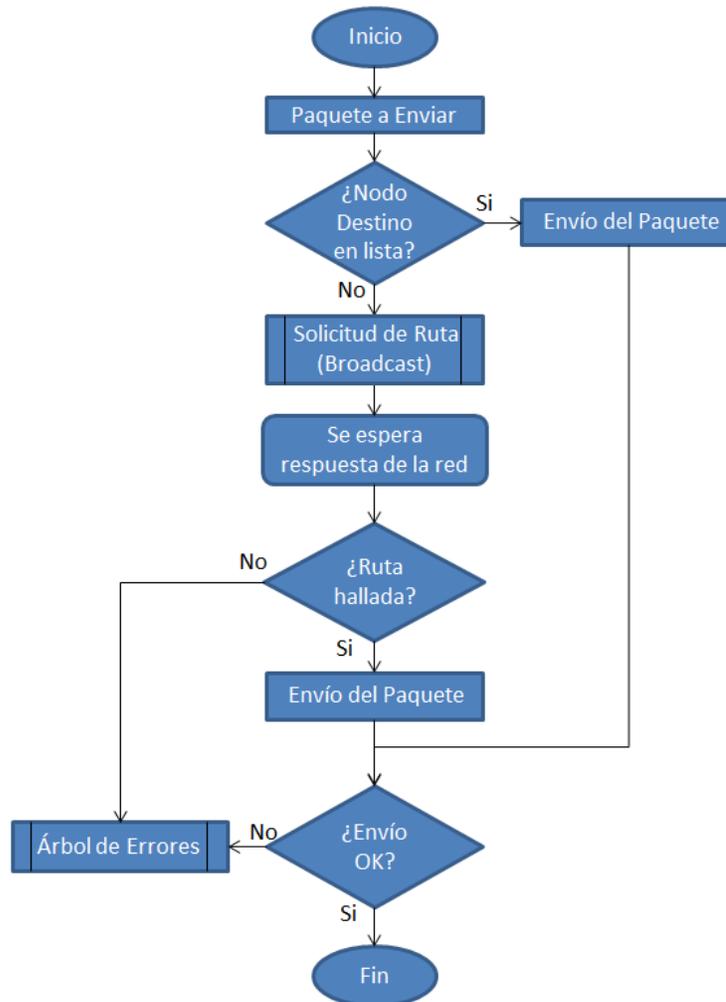


Figura 4.70: Proceso de solicitud de rutas

#### 4.4.6. Algoritmo Recepción de Solicitud de Ruta

Cuando un dispositivo recibe una solicitud de ruta lo primero que se ha de revisar es si la solicitud esta dirigida al mismo, puesto que de ser así solo debe esperar un tiempo, que ha sido fijado en un segundo para poder recibir todas las posibles rutas y escoger la que represente un menor costo energético, a través de la cual se enviaría la aceptación de ruta, pero de no ser ese el caso, entonces el dispositivo debe revisar si se puede seguir re-enviando la solicitud, esto se determina con el octeto Radius ubicado en la capa de red, de no ser posible reenvío de paquetes se debe informar que por esa ruta no es posible el proceso, en

caso contrario se procede a añadir la ruta como una posible solución, posteriormente se ha de revisar si el dispositivo destino se encuentra en la zona cercana, puesto que de ser así se debe enviar la solicitud dirigida a este, pero si no se encuentra se debe revisar si ya se ha enviado una solicitud que involucre a ambos dispositivos, el originador y el destino, esto se debe a que si no se ha enviado previamente se procede a re-enviar como broadcast decrementando previamente el Radius, en el caso contrario se debe reenviar decrementando el Radius si y solo si la nueva ruta presenta un menor costo energético. Este proceso se puede observar en la Figura 4.71.

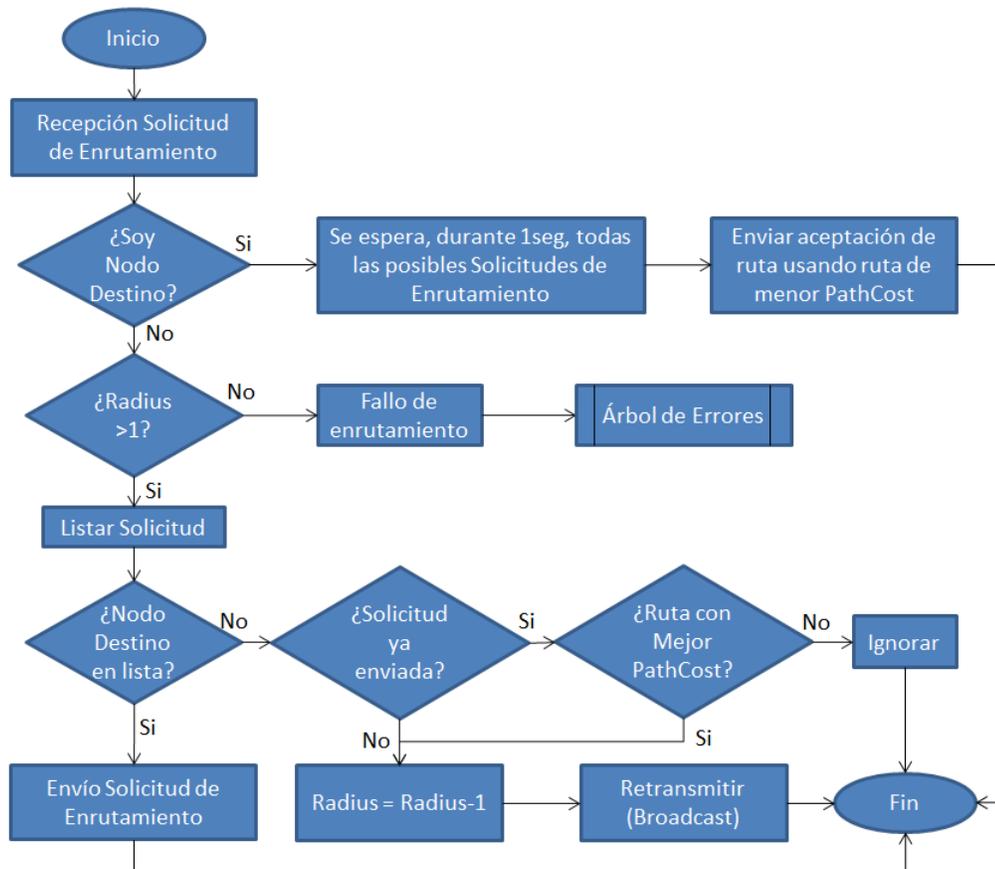


Figura 4.71: Proceso de recepción de solicitud de ruta

#### 4.4.7. Algoritmo Aceptación de Ruta

Si un dispositivo recibe una aceptación de ruta, esta debe de estar dirigida a para él mismo, entonces se procede a añadir a la lista la ruta en dirección al dispositivo destino, usando al dispositivo que le envió la aceptación de ruta como siguiente hoc en la ruta, sin embargo si todavía no se ha llegado hasta el originados de la solicitud de ruta se debe añadir a la lista de ruta en dirección al dispositivo originador de la solicitud usando el camino que genere menor costo energético y se reenvía por esa ruta la aceptación de la ruta, si en efecto la aceptación la recibe el originador de la solicitud de la ruta se ha completado el proceso de enrutamiento y se procede a enviar el paquete pendiente. Este proceso se puede observar en

la Figura 4.72.

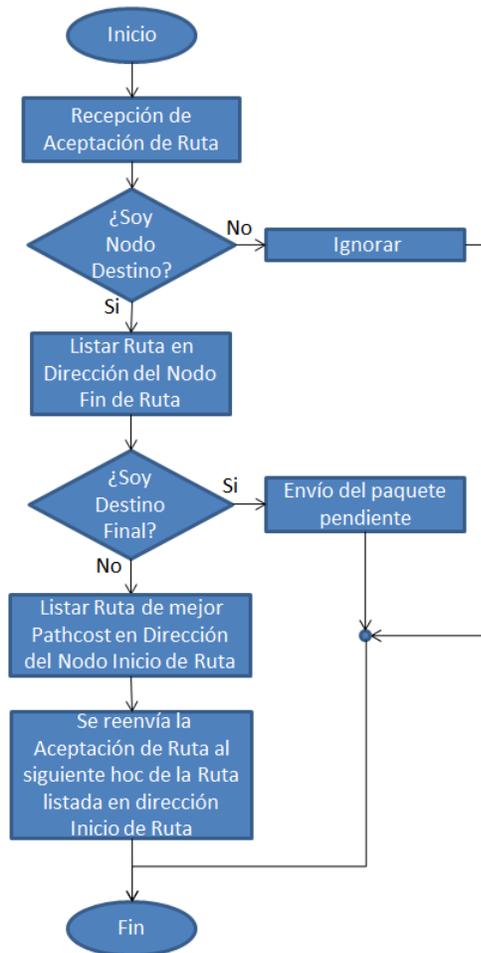


Figura 4.72: Diagrama de bloques del algoritmo de Aceptación de Ruta.

#### 4.4.8. Algoritmo de Medición de Baterías

En las redes de baja transferencia se presta la utilización de dispositivos con alimentación a base de celdas o baterías por su bajo consumo de potencia. Esto hace importante un mecanismo que permita determinar en que estado de carga se encuentran estas permitiendo su reemplazo antes de estas estar por debajo de la tensión necesaria y provocando innecesarias salidas de operación de nodos de la red por fallo de energía.

Como se puede observar en la Figura 4.73, para realizar este proceso se debe, primeramente, activar la tensión de referencia interna del PIC18F27J53, de aproximadamente 1,2v para poder medirla con el conversor analógico-digital, el resultado de esta medición es extrapolado hasta que el conversor este con a full escala, el resultado de dicha operación indica el valor de tensión del sistema de alimentación, posteriormente se desactiva la tensión de referencia interna del PIC para ahorrar energía.

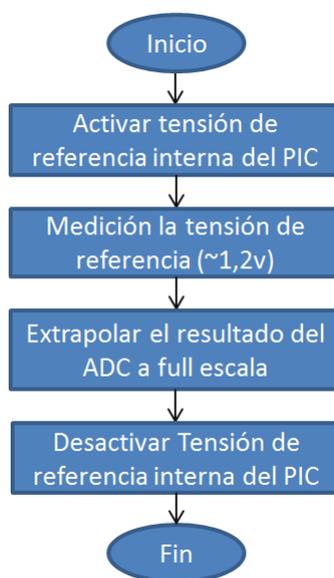


Figura 4.73: Diagrama de bloques del algoritmo de medición de batería.

#### 4.4.9. Algoritmo de Listado de Errores

En toda red existen fallos de comunicación por motivos conocidos como pérdida de comunicación con un nodo, no fue posible establecer una ruta, entre otros; o por motivo del mal-funcionamiento de los procesos internos del código. Dentro del diseño de la red presentada en este trabajo de grado se consideraron tres fallos básicos de comunicación como los son:

**Error por falla en la comunicación con un dispositivo cercano** Este error ocurre cuando se desea transmitir un paquete a un nodo que se encuentra en la lista de nodos cercanos pero se hace imposible la conexión y por ende la transmisión de dicho paquete.

Si este error se origina dentro de los procesos de la red, esta última debe eliminar al destino de la comunicación de la lista de dispositivos cercanos e intentar re-establecer la comunicación a través de una ruta como se muestra en la Figura 4.74. Seguidamente se debe listar el error del código con fecha y hora del momento en que ocurrió.

**Error por falla en la comunicación a través de una ruta** Este error se origina cuando se desea transmitir un paquete a un nodo que se encuentra en la lista de rutas pero se hace imposible la entrega del paquete, por estar fuera de funcionamiento algún nodo intermedio o el nodo de destino.

Si se este error ocurre, el sistema debe responder eliminando la ruta al dispositivo e intentar re-establecer la comunicación a través de una nueva ruta como se muestra en la Figura 4.74. Seguidamente se debe listar el código del error con fecha y hora del momento en que ocurrió.

**Error durante el proceso de enrutamiento** Este error se produce cuando se hace imposible la ubicación de un dispositivo dentro de la red a pesar de haber realizado las respectivas peticiones de ruta.

Si se produce este error o cualquiera de los antes mencionados se debe incluir el error con su respectivo código y fecha en el que ocurrió en la lista de errores.

Este proceso se puede observar en la Figura 4.74.

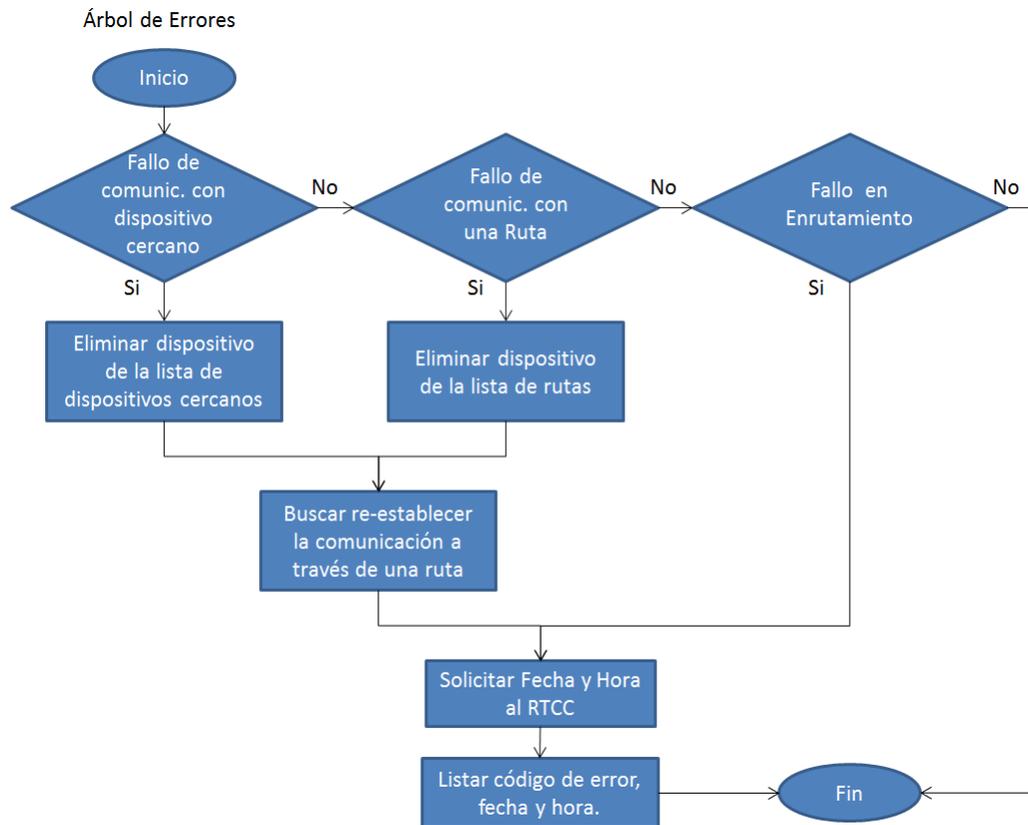


Figura 4.74: Diagrama de bloques del algoritmo de listado de errores.

## 4.5. Desarrollo de la Interfaz

Esta sección tiene como objetivo la creación de la interfaz de usuario que permite la configuración de los parámetros de operación de los sensores y observación de algunos parámetros de interés propios de la red.

### 4.5.1. Depurador de errores

Durante el proceso de desarrollo del circuito y el código de comunicaciones inalámbricas surgió la necesidad de observar los paquetes que estaban siendo transmitidos, si estos estaban estructurados de la forma apropiada y si el sistema respondía como era esperado, entonces se procedió al desarrollo de una interfaz comunicada a través del estándar USB y usando el

entorno de desarrollo Microsoft Visual C# 2010, en la cual se mostraba cada paquete recibido por un dispositivo conectado, así como cada paquete enviado por el mismo, cada paquete es identificado por la interfaz, determina su origen y fin, el tipo de paquete que es y mantiene un registro de todos los paquetes enviados y recibidos.

La interfaz también permite el envío de datos al dispositivo conectado, con lo que también fue usado para corroborar el correcto funcionamiento de cada comando establecido para el sistema.

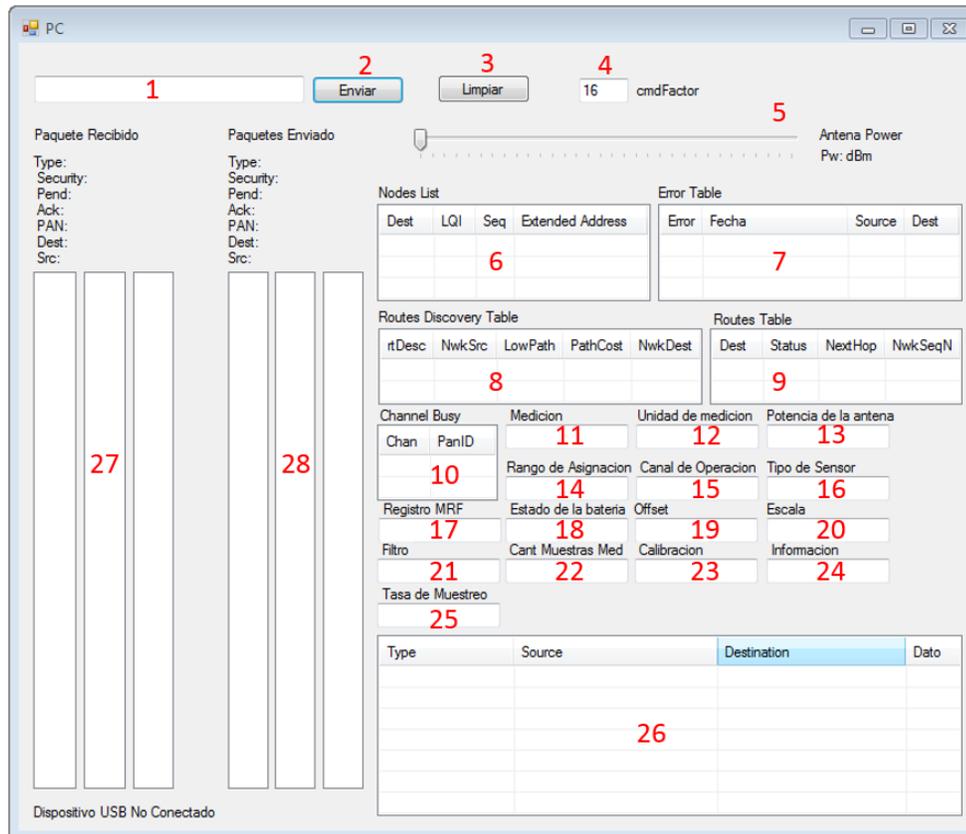


Figura 4.75: Depurador de errores

El depurador, mostrado en la Figura 4.75, cuenta con un espacio de texto para escribir la data al dispositivo conectado (1), esta debe ser escrita en hexadecimal en dos dígitos cada byte y los bytes deben estar completos, siguen la estructura de los comandos establecidos, esto es, 2 bytes indicando el identificador corto de 16bits del dispositivo al cual se va a enviar la data seguido del comando, con su *payload* de ser necesario, así mismo cuenta con un botón (2) para enviar el texto escrito al dispositivo conectado y un botón (3) que limpia todos los campos de la interfaz.

La interfaz también incluye una barra móvil para ajustar rápidamente la potencia de transmisión del dispositivo (5), una tabla de nodos cercanos (6), una tabla de rutas (7), una tabla de posibles rutas (8), una tabla de errores (9), una tabla de canales ocupados (10), un recuadro para mostrar la medición (11), un recuadro para mostrar su unidad (12), un recuadro para mostrar la potencia de transmisión (13), un recuadro para mostrar el rango de direcciones

posibles a asignar (14), un recuadro para mostrar el canal en el que se esta operando (15), un recuadro para mostrar el tipo de sensor (16), un recuadro para mostrar el contenido de un registro del modulo de comunicación inalámbrica (17), un recuadro para mostrar el estado de la batería (18), un recuadro para mostrar el offset (19), un recuadro para mostrar la escala de la medición (20), un recuadro para mostrar el tipo de filtro digital aplicado (21), un recuadro para mostrar la cantidad de mediciones tomadas para ser promediadas (22), un recuadro para mostrar los registros de calibración (23), un recuadro para mostrar los registros de información (24), un recuadro para mostrar el intervalo de tiempo entre muestras (25), tres recuadros para mostrar el ultimo dato transmitido inalámbricamente (27) y tres recuadros para mostrar el ultimo dato recibido inalámbricamente (28).

Cuando la respuesta del comando, si este tiene una, el depurador automáticamente la identifica, usando la información ubicada en el recuadro para el corrimiento de los comandos (4) y la despliega en el recuadro correspondiente mientras registra todos los datos transmitidos inalámbricamente en el registro (26).

## 4.5.2. Interfaz

### 4.5.2.1. Desarrollo

Se desarrollo una interfaz orientada a mostrar al usuario la medición de los dispositivos, así como la información adicional de cada dispositivo. esta consta de 3 ventanas.

La ventana principal, mostrada en la Figura 4.76, muestra de forma resumida el estado del sensor, lo que se limita a su nombre propio (1), medición con unidad (3), el estado de la batería (4) y la muestra de los errores ocurridos(5), así como la información de los desarrolladores (2).

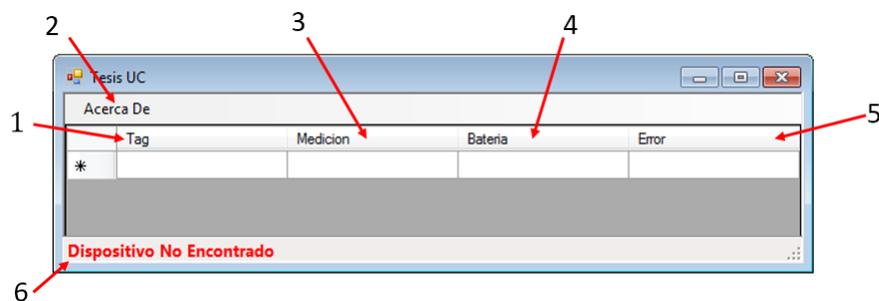


Figura 4.76: Vista de mediciones sin listas de errores

Los errores se muestran en una lista desplegable (1), ese se puede observar en la figura 4.77.

La interfaz esta a la espera de la adición de algún dispositivo adicional a la red para añadirlo a la lista y poder desplegar su información de medición, así mismo si algún dispositivo es extraído de la red o se genera algún fallo de comunicación con el mismo, el nodo es eliminado de la lista, de modo que solo se muestren los dispositivos conectados satisfactoriamente a la red.

Si se desea observar toda la información del dispositivo, se despliega una ventana que mostrara dicha información, ver figura 4.78, la cual muestra el nombre propio del sensor (1),

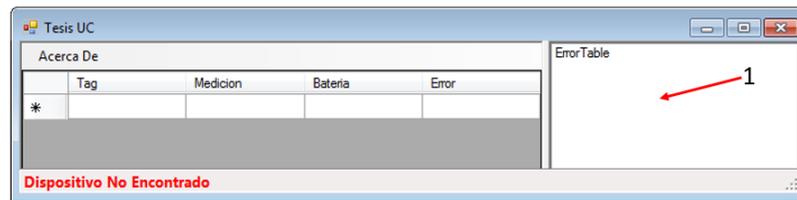


Figura 4.77: Vista de mediciones con listas de errores

su tipo (2), la unidad de medición (3) el offset (4), la escala (5), los registros de calibración (6) y (7), la cantidad de muestras a ser promediada (8), el intervalo entre muestras (9), el filtro digital aplicado (10), los registros del información (11), (12), (13), (14) y (15) y un botón destinado a guardar los cambios en el dispositivo (16).

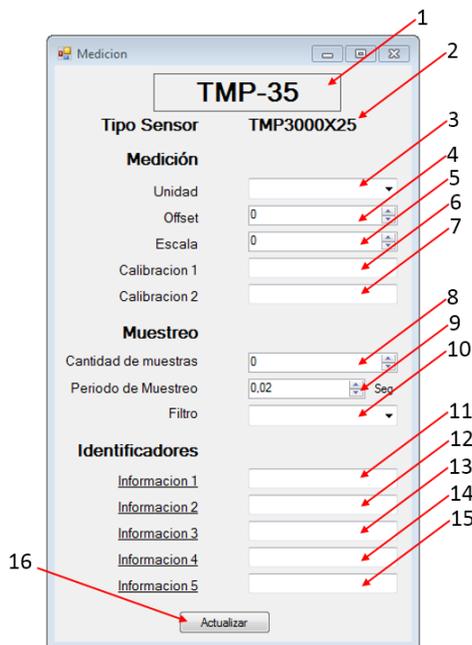


Figura 4.78: Vista de Información adicional

## 4.6. Pruebas

### 4.6.1. Alcance

Se realizaron una serie de pruebas para determinar la máxima distancia de transmisión sin que ocurriesen fallos de comunicación. Para esto se estableció una red usando el nodo principal con el fin de, posteriormente, realizar una comunicación con un nodo.

Los dispositivos se ubicaron de tal forma que la máxima radiación apuntase entre ellos y el nodo se fue alejando en pasos regulares de un(1) metro, mientras se realizaban transmisiones para verificar el enlace. Esto se realizo hasta que la conexión fallo y el nodo fue expulsado de la red debido al fallo de la comunicación.

Esta prueba se realizó en dos ambientes de trabajo:

- a) La primera prueba se realizó en un entorno cerrado, en donde los nodos no se encontraban en línea de visión, dando como resultado un alcance máximo de 13m antes de que la conexión fallara
- b) La segunda prueba se realizó al aire libre, con visión directa entre el transmisor y el receptor, y con la menor cantidad de interferencias posibles, dando como resultado 108m, siendo esto lo esperado, ya que el módulo de comunicaciones especifica un alcance de hasta 100m.

#### 4.6.2. Consumo de potencia

El equipo fue puesto a prueba en diferentes condiciones de uso para determinar su consumo en cada caso y poder estimar el tiempo de vida útil en caso de operar a batería.

Los estados en los cuales se probaron los dispositivos fueron:

- a) En transmisión continua, la cual se estableció como una comunicación cada 10ms, el consumo de los dispositivos resultó en el orden de los 40mA.
- b) En completo reposo se midió un consumo de alrededor de 39mA.
- c) Durante el proceso de medición se obtuvo un consumo de 39,6mA.

Por lo tanto si el sistema llegase a funcionar con baterías de una capacidad de 2400mAh, podría trabajar de forma continua aproximadamente 60h.

# Capítulo 5

## Conclusiones

Se logró una completa revisión del estándar IEEE 802.15.4, lo que permitió el manejo de la Capa Física (PHY) y la Capa de Control de Acceso al Medio (MAC), desglosando su estructura protocolar PPDU y MPDU para la Capa PHY y la MAC respectivamente, en sus componentes básicas: SHR, PHR y PHY *payload* de la capa PHY y MHR, MAC *payload* y MFR de la MAC; así como el manejo de los subcampos que las componen dentro de los que se puede mencionar el Frame Control, de gran importancia para la transmisión de datos por contener la indicación de la información contenida dentro de cada paquete. Se revisaron los diferentes tipos de tramas manejadas por la MAC para el envío de paquetes como lo son: *beacon*, *command*, *data* y *acknowledge* así como el desarrollo de los procesos de asociación a la red, proceso que es controlado por la capa MAC. Todo esto permitió el desarrollo de un código para el microcontrolador PIC18F27J53 que, regido por el estándar IEEE 802.15.4, brindara el soporte para las capas de red y de aplicación. De igual forma esto logró el entendimiento en su totalidad del funcionamiento interno del módulo de comunicación inalámbrica MRF24J40MA logrando su exitosa configuración y una efectiva comunicación entre los nodos de la red.

Se determinó que el protocolo óptimo para el manejo de las capas superiores, fue ZigBee por proveer a la red de una capacidad de manejo de alrededor de 64000 nodos, sin límite de coordinadores de red y rutas de comunicación de indefinido número de saltos, convirtiéndola en una red con un área de cobertura muy grande aún utilizando dispositivos de relativo corto alcance.

Se diseñaron, fabricaron y programaron satisfactoriamente los dispositivos correspondientes a un Nodo Principal, dos Nodos Sensores con ADC simple, un Nodo Sensor con ADC Diferencial, un Sensor de Temperatura, un Sensor de Presión y una Tarjeta para Pruebas de Red. Dichos dispositivos son capaces de automáticamente aperturar una red, asociar dispositivos, enviar mensajes inteligentemente, buscar rutas alternas en caso de fallo, entre otros. Para esto fue necesaria la selección del módulo de comunicación inalámbrica compatibles con el estándar IEEE 802.15.4, dentro de los que fue seleccionado el MRF24J40MA sobresaliente por su bajo costo y capacidad para manejo de topología Mesh; De igual forma fue necesaria la selección del microcontrolador, resultando el PIC18F27J53 como el dispositivo más pequeño disponible capaz de cubrir todos los requerimientos establecidos para su selección.

Se desarrolló satisfactoriamente el algoritmo destinado a brindarle la capacidad al microcontrolador de soportar los procesos comprendidos dentro del estándar IEEE 802.15.4 y el protocolo de la Capa de Red, algoritmo que le brinda a la red, la capacidad de aperturar

una PAN, asociar dispositivos a la red, búsqueda de rutas de forma inteligente, listar errores, enviar información, medición de baterías, entre otros.

De igual forma se desarrolló el protocolo para la capa de aplicación el cual permitió la comunicación de la interfaz con la red surtiendo a dicha capa con una lista de 55 comandos que permite la solicitud y configuración tanto de los parámetros del sensor como los de la red. Además provee de mecanismos de *reset* por software, haciendo posible el reinicio de la red a distancia.

El desarrollo de los aspectos antes mencionados dan soporte al microcontrolador PIC18F27J53 para el manejo del modulo de comunicaciones MRF24J40 y el protocolo ZigBee, puesto que el *stack*, de dicho protocolo, que provee la empresa microchip ofrece muy poco soporte para los dispositivos de la gama 18.

Se logró, de forma exitosa, el desarrollo de la interfaz que permite al usuario el acceso a la información de los nodos de la red como la configuración de parámetros propios de red como por ejemplo el ajustes de canal, potencia a la salida a la base de la antena entre otros.

Se verificó el correcto funcionamiento de la interfaz, el algoritmo del microcontrolador y el protocolo de comunicaciones mediante un conjunto de pruebas preestablecidas certificando la confiabilidad de la red en la totalidad de los tramos que la componen.

Finalmente se concluye que el cableado comúnmente utilizado para la interconexión de redes de sensores puede ser fácilmente reemplazado por una LR-WPAN, disminuyendo el costo de instalación, materiales y mano de obra que esto requeriría. Se demostró la eficacia de la red mediante la implementación de algoritmos de verificación de correcta recepción o *acknowledge*. Además de procesos de CRC proporcionados por el mismo módulo de comunicación inalámbrica convirtiéndola además en una red de alta confiabilidad.

# Recomendaciones

- Como continuación de este trabajo de grado, se propone el rediseño del hardware de los dispositivos desarrollados, por dispositivos de mucho menor tamaño y mucho más compactos, estudiando la remoción del plano de tierra y los efectos electromagnéticos que ésto pudieran ocasionar, además del desarrollo de algoritmo que permita la encriptación y desencriptación de paquetes haciendo uso de los módulos contenidos en el MRF24J40MA para este fin.
- Como continuación de este trabajo de grado, se propone el desarrollo del código para añadir la opción de configuración del modo de operación de la red bajo el modo *beacon enable* incluyendo transmisión bajo supertrama, con el objetivo de lograr colocar al equipo, microcontrolador y módulo de comunicaciones inalámbricas, a operar en su modalidad de bajo consumo y así alargar el tiempo de vida útil del trabajo a batería.
- Realizar un estudio del retardo existente en las transmisiones a lo largo de la red y de la factibilidad de usar los dispositivos como elementos de control.

# Bibliografía

- [1] Creed Huddleston, *Intelligent Sensor Design Using the Microchip dsPIC*, Elsevier, Ed. Newnes, 2007. 1.1
- [2] BCC Reasearch [En línea]. (2011, Marzo) <http://www.bccresearch.com/market-research/instrumentation-and-sensors/sensors-technologies-markets-ias006d.html>. 1.1
- [3] ZigBee Alliance, *ZigBee Specification*, 2004. 2.1
- [4] Tim Cutler, “Implementing zigbee wireless mesh networking,” 2005. 2.1
- [5] Fred Eady, *Hands on ZigBee Implementing 802.15.4 with Microcontrollers*, Newness, Ed. Elsevier, 2007. 2.1, 2.2.2.1
- [6] Dogan Ibrahim, *Advanced PIC Microcontroller Projects in C (From USB to RTOS with the PIC18F Series)*, Elseiver, Ed. Newnes, 2008. 2.1
- [7] Br. Franco Caverzan / Br. Carlos Vargas, “Desarrollo de un Sistema de Monitoreo Inalámbrico del Nivel en Tanques de Almacenamiento de Líquidos Basados en Dispositivos Compatibles con el Estándar IEEE 802.15.4,” Master’s thesis, Escuela de Eléctrica - Facultad de Ingeniería - Universidad de Carabobo - Venezuela, 2011. 2.1
- [8] Behrouz A. Forouzan, *TCP/IP Protocol Suite*, McGrawHill, Ed. Higher Education, 2006. 2.2.1, 2.2.1.1, 2.2.1, 2.2.1.2, 2.2.1.4, 2.2.1.5, 2.2.1.6, 2.2.1.7
- [9] IEEE 802.15 Working Group, *802.15.4 Standard - Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, I. C. Society, Ed. Institute of Electrical and Electronic Engineers, 2003. 2.2.2, 1, 2.2.2.1, 2.2.2.1, 2.2.2.2, 2.2.2.3, 2.6, 2.7, 2.8, 2.9, 2.1, 2.10, 2.11, 2.12, 2.2, 2.3, 2.13, 2.14, 2.15, 2.4, 2.16, 1, 2.18, 2.19, 2.20, 2.21, 2.5
- [10] Microchip Technology Inc., *AN1204 Microchip MiWi P2P Wireless Protocol*, 2010. 2.3, 2.4
- [11] —, *AN1066 MiWi Wireless Networking Protocol Stack*, 2007. 2.5
- [12] —, *MRF24J40 Data Sheet IEEE 802.15.4 2.4 GHz RF Transceiver*, 2010. a), b), c)
- [13] Drew Gislason, *ZigBee Wireless Networking*, Newnes, Ed. Elseiver, 2005. 1

- [14] *Guidelines for 64-bit Global Identifier (EUI-64) (2012)*, IEEE Standards Association, Std. 1, 2.2.5.6
- [15] Zigbee Alliance, *ZigBee Specification*, Junio 2005. 2.22, 2.6, 2.7, 2.23, 2.24, 2.25, 2.8, 2.26, 2.9, 2.27, 2.28, 2.10, 2.29
- [16] Atmel Corporation, *ZigBit 2.4GHz Wireless Modules ATZB-24-A2/B0 Datasheet*, 2009. 2.32
- [17] Digi International Inc., *XBee & XBee-PRO OEM RF Module*, Junio 2012. 2.35
- [18] Microchip Technology Inc., *MRF24J40MB Data Sheet*, 2009. 2.40
- [19] ———, *MRF24J40MA Data Sheet*, 2008. 2.44
- [20] Maritza Barrios Yaselli, *Manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales*, A. J. de Castillo, Ed. FEDUPEL, Reimpresión-2006. 3.1
- [21] Jacqueline Hurtado, *Metodología de la Investigación - Guía para la comprensión holística de la ciencia*, E. Q. SA, Ed. CIEA-Sypal, 2010. 3.2
- [22] Microchip Product Selector Tool [En línea]. (2012, Julio) [www.microchip.com/productselector/mcuproductselector.html](http://www.microchip.com/productselector/mcuproductselector.html). 4.2.1.2
- [23] National Semiconductor, *3-Terminal Adjustable Regulator*, Febrero 2011. 4.2.1.4
- [24] Q. J. Ballesteros María, Melo Harvey, “Sistema De Transmisión Inalámbrica De Señales ECG Y De Temperatura Para Ambientes Hospitalarios (SINH0),” *Revista de Ingeniería Biomédica de la Escuela de Ingeniería de Antioquia*, vol. 4, p. 1a7, 2010.
- [25] Computer Simulation Technology, *Microwave Studio 2010 Help*, 2010.

# Glosario

- BI (Beacon Interval): tiempo entre Beacons para un modo de operación *beacon enable*.
- BSN (Beacon Sequence Number): número que incrementa en "uno" con cada Beacons para denotar su orden o secuencia.
- CAP (Contention Access Period) : tiempo designado para la transmisión de información dentro de una supertrama.
- CCA (Clear Channel Assessment): nivel de umbral para la detección de canal ocupado.
- CRC (Cyclic Redundancy Check): verificación de redundancia cíclica.
- CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance): algoritmo para detección de disponibilidad de canal libre.
- ED (Energy Detect) : mecanismo para detección de energía a la entrada de la antena.
- FFD (Full Function Device): dispositivo con set de funciones extendido.
- GTS (Guaranteed Time Slot): bloque de tiempo asegurado
- LQ (Link Quality): calidad de enlace
- LQI: (Link Quality Indication): indicador de calidad de enlace.
- LR-WPAN( Low Rate - Wireless Personal Area Network): red inalámbrica de área personal de baja transferencia.
- LSB (Less Significant Bit): bit(s) menos significativo(s).
- MAC (Medium Access Control): capa de control de acceso al medio.
- MFR ( MAC Footer) : pie de MAC
- MHR (MAC Header): encabezado de MAC.
- MSB (Most Significant Bit): bit más significativo.
- MPDU (MAC Protocol Data Unit): nombre que se le da a la estructura general de la MAC.

- MSDU (MAC Service Data Unit): nombre que se le da a la estructura del *payload* de la MAC.
- O-QPSK(Offset Quadrature Phase-Shift Keying) : tipo de modulación que utiliza el MRF24J40MA.
- OSI (Open System Interconetion): Sistemas de Interconexión Abierta.
- PAN (Personal Area Network): red de área personal.
- PDU (Protocolar Data Unit): unidad protocolar básica.
- PHR (PHY Header) : encabezado de la capa física.
- PHY : capa física
- PPDU (PHY Protocolar Data Unit) : estructura general de la capa física.
- PSDU (PHY Service Data Unit) : estructura básica del *payload* de la capa física.
- RF ( Radio Frequency) : Radio frecuencia.
- RFD (Reduced Function Device) : dispositivo de funciones reducidas.
- RSSI (Received Signal Strength Indication): indicador de intensidad de enlace.
- RX (Receive - Receiver) : recepción / receptor.
- SFD (Start-of-Frame Delimiter) : delimitador de inicio de trama.
- SHR (Synchronization Header) : encabezado de sincronización.
- TRX (Transceiver) : transreceptor
- TX (Transmit - Transmitter) : transmitir / transmisor.
- WPAN (Wireless Personal Area Network) : red inalámbrica de área personal.

# Anexos

# Anexo A

## Diagramas funcionales

En la Figura 5.1 se muestra el diagrama funcional del coordinador de la red.

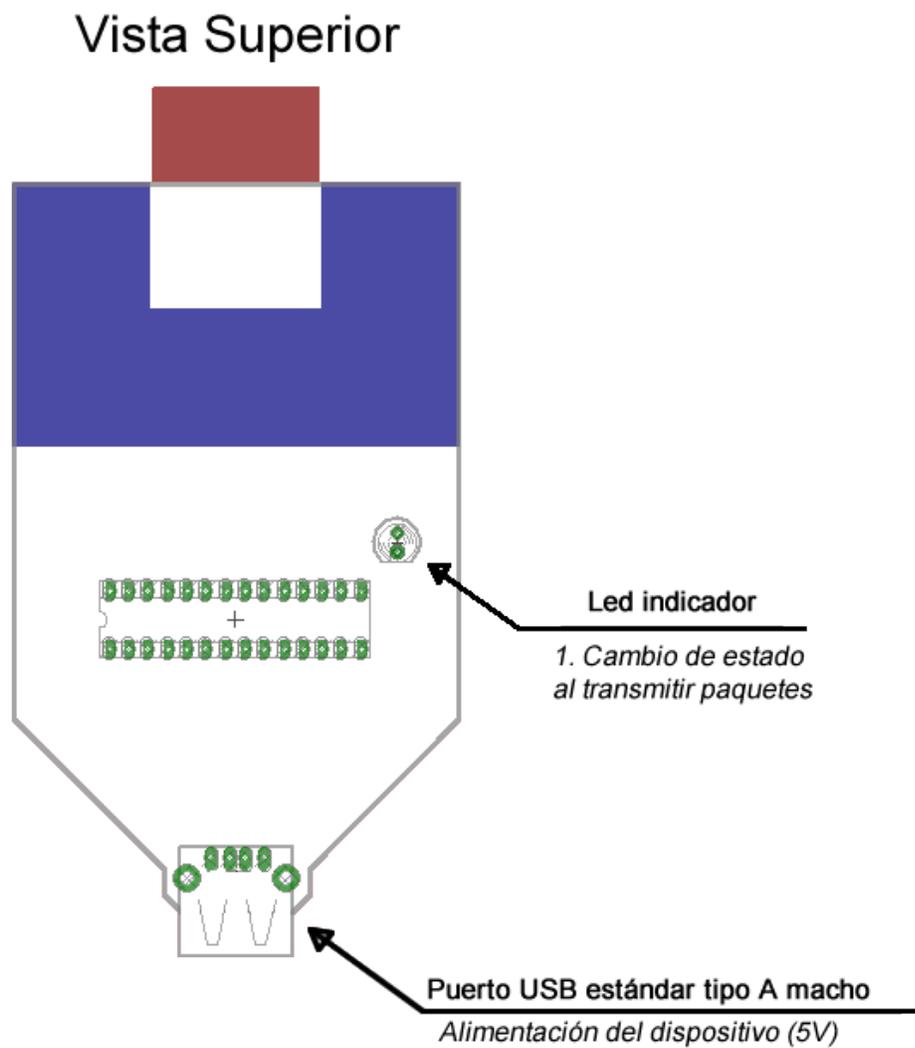


Figura 5.1: Diagrama funcional del coordinador de la red

En la Figura 5.2 se muestra el diagrama funcional de los nodos.

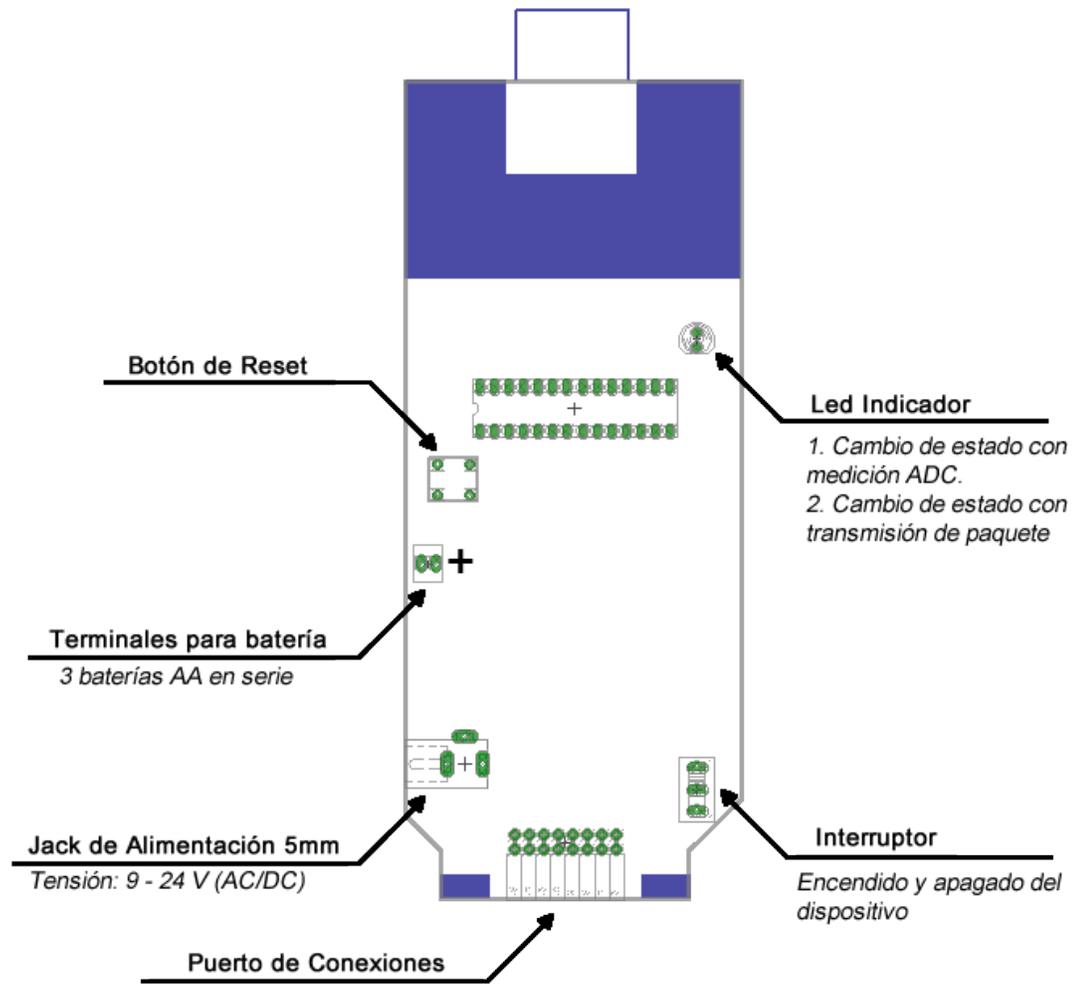


Figura 5.2: Diagrama funcional de los nodos

En la Figura 5.3 se muestra la distribución del puerto de conexiones del nodo con convertor analógico-digital simple.

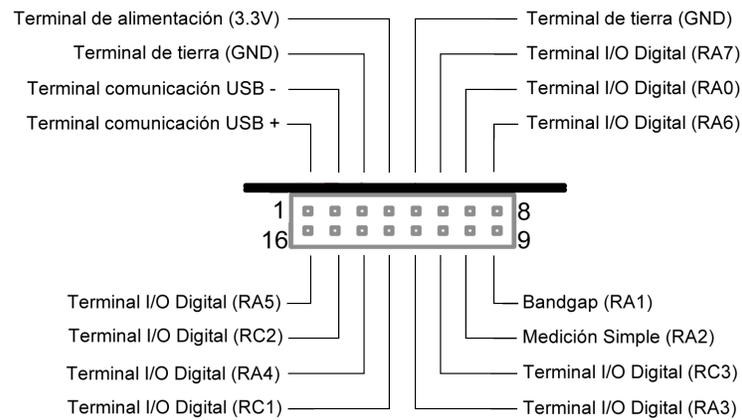


Figura 5.3: Puerto de conexiones del nodo con convertor analógico-digital simple

En la Figura 5.4 se muestra la distribución del puerto de conexiones del nodo con convertor analógico-digital diferencial.

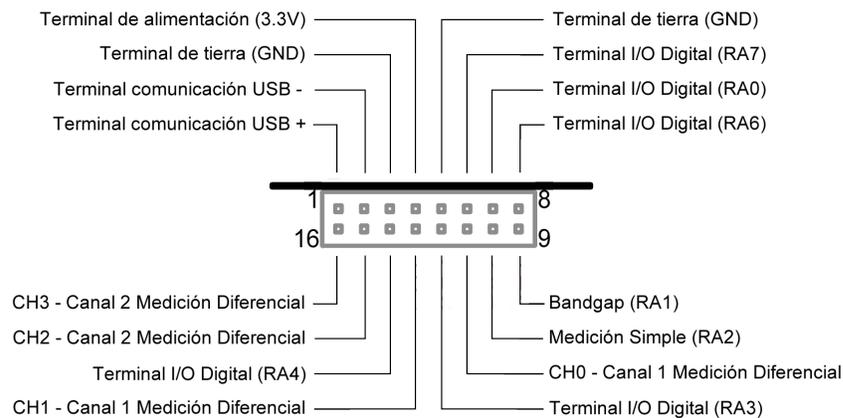


Figura 5.4: Puerto de conexiones del nodo con convertor analógico-digital diferencial

# Anexo B

## Pruebas realizadas al protocolo de la capa de aplicación

A continuación se procederán mostrar los resultados obtenidos de las pruebas realizadas al los comandos del protocolo de la capa de aplicación, las cuales fueron realizadas en un ambiente cerrado y usando una computadora para ejecutar el depurador desarrollado, la cual posee un sistema operativo Microsoft Windows 7 de 64 bits.

Las pruebas fueron realizadas con una potencia de transmisión de -10dBm y ubicando los dispositivos de la siguiente forma:

- El coordinador de la red permaneció conectado a través del terminal usb a la computadora, desde la cual se transmitieron los comandos.
- Un dispositivo ubicado a cuatro(4) metros del coordinador de la red, sin obstáculos de por medio.
- Un dispositivo que se encuentra fuera del alcance del coordinador de la red, ubicado a ocho(8) metros del mismo y a cuatro(4) metros del dispositivo anterior.