

# TRABAJO ESPECIAL DE GRADO

## DISEÑO DE UNA RED INDUSTRIAL MODBUS RTU INALÁMBRICA

Presentado ante la Ilustre  
Universidad Central de Venezuela  
por el Br. Alejandro J. Temprano G.  
para optar al título de  
Ingeniero Electricista

Caracas, 2017

# TRABAJO ESPECIAL DE GRADO

## **DISEÑO DE UNA RED INDUSTRIAL MODBUS RTU INALÁMBRICA**

Tutor Académico: Prof. José Alonso

Presentado ante la Ilustre  
Universidad Central de Venezuela  
por el Br. Alejandro J. Temprano G.  
para optar al título de  
Ingeniero Electricista

Caracas, 2017


## CONSTANCIA DE APROBACIÓN


Caracas, 17 de noviembre de 2017


Los abajo firmantes, miembros del Jurado designado por el Consejo de Escuela de Ingeniería Eléctrica, para evaluar el Trabajo Especial de Grado presentado por el Br. Alejandro J. Temprano G., titulado:

“DISEÑO DE UNA RED INDUSTRIAL MODBUS RTU INALÁMBRICA”

Consideran que el mismo cumple con los requisitos exigidos por el plan de estudios conducente al Título de Ingeniero Electricista en la opción de Electrónica y Control, y sin que ello signifique que se hacen solidarios con las ideas expuestas por el autor, lo declaran APROBADO.

  
Prof. Servando Álvarez  
Jurado

  
Prof. Jesús Sánchez  
Jurado

  
Prof. José Alonso  
Tutor Académico

## DEDICATORIA

A mi papá.

## RECONOCIMIENTOS Y AGRADECIMIENTOS

Al profesor José Alonso por su apoyo durante todo el proyecto. A mis familiares y amigos por estar ahí cuando los necesité. A Fabricio, a Ezequiel y a Omar por toda la ayuda y la paciencia. Sin todos ellos este proyecto no hubiera salido adelante.

Muchas gracias.

**Alejandro J. Temprano G.**

**DISEÑO DE UNA RED INDUSTRIAL MODBUS RTU  
INALÁMBRICA**

**Tutor Académico: Prof. José Alonso. Tesis. Caracas. U.C.V. Facultad de Ingeniería. Escuela de Ingeniería Eléctrica. Ingeniero Electricista. Opción Electrónica y Control. Institución: U.C.V. Facultad de Ingeniería. Escuela de Ingeniería Eléctrica. 2017. Trabajo de Grado. Pág. 67 + anexos.**

**Palabras claves:** Red industrial inalámbrica, Radiofrecuencia, Transmisión inalámbrica, MODBUS RTU, PIC, RFM69HW, RFM69HWC, SX1231, SX1231h.

**Resumen.** En el siguiente trabajo se presenta la implementación de una red industrial MODBUS RTU inalámbrica mediante el diseño de un dispositivo de puerta de enlace. En primera instancia se presentó una breve investigación sobre el protocolo MODBUS RTU y algunas de las tecnologías de transmisión inalámbrica más utilizadas, posteriormente se seleccionó la tecnología inalámbrica más adecuada y se planteó el hardware del dispositivo de puerta de enlace. Finalmente se diseñó el firmware del dispositivo y se ejecutaron las pruebas para validar el diseño planteado realizando el análisis de propagación del dispositivo y una implementación en condiciones reales.

# ÍNDICE GENERAL

	Pág.
CONSTANCIA DE APROBACIÓN.....	iii
DEDICATORIA .....	iv
RECONOCIMIENTOS Y AGRADECIMIENTOS .....	v
RESUMEN.....	vi
ÍNDICE GENERAL.....	vii
ÍNDICE DE TABLAS .....	x
ÍNDICE DE FIGURAS Y/O GRÁFICOS .....	xi
SIGLAS Y ACRÓNIMOS.....	xiii
<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO I.....</b>	<b>4</b>
<b>1. MARCO REFERENCIAL.....</b>	<b>4</b>
1.1 Justificación e importancia.....	4
1.1 Objetivo general .....	5
1.2 Objetivos específicos .....	5
<b>CAPÍTULO II .....</b>	<b>6</b>
<b>2 MARCO TEÓRICO.....</b>	<b>6</b>
2.1 Redes de comunicación.....	6
2.2 Tecnologías inalámbricas de transmisión de datos .....	9
2.3 Modelo de referencia OSI.....	11
2.4 El protocolo MODBUS.....	16
2.4.1 MODBUS en la capa de enlace.....	16
2.4.1.1 MODBUS RTU.....	18
2.4.2 MODBUS en la capa física.....	20

<b>CAPÍTULO III</b> .....	21
3. PROPUESTA Y REQUERIMIENTOS DEL SISTEMA .....	21
3.1 Propuesta del sistema .....	21
3.2 Requerimientos del dispositivo .....	22
<b>CAPÍTULO IV</b> .....	24
4. DESCRIPCIÓN DEL HARDWARE.....	24
4.1 Descripción general del hardware .....	24
4.2 El módulo de comunicación inalámbrica .....	25
4.3 Módulo de control .....	27
4.4 Interfaz RS485 .....	28
4.5 Fuente de alimentación .....	28
<b>CAPÍTULO V</b> .....	29
5. DISEÑO DEL FIRMWARE.....	29
5.1 Descripción general del firmware .....	29
5.2 Configuración inicial del microcontrolador .....	31
5.3 Configuración inicial del radio.....	33
5.3.1 Función RFM69HW_Config().....	33
5.3.2 Función RF_ReConfig() .....	36
5.3.3 Función RF_Mode_ChangeTo() .....	36
5.3.4 Función RF_Send() .....	37
5.3.5 Función RF_PacketRecived() .....	38
5.3.6 Función RF_ReadRSSI() .....	38
5.3.7 Función RF_Encrypt() .....	39
5.3.8 Función RF_Debug() .....	39
5.3.9 Definiciones y valores iniciales .....	40
5.4 Estado 0: Espera.....	40
5.5 Estado 1: Recepción por el RS485.....	41
5.6 Estado 2: Procesado de la trama.....	43
5.7 Estado 3: Recepción por el radio .....	45
5.8 Estado 4: Envío por el radio.....	47



<b>CAPÍTULO VI</b> .....	50
6. VALIDACIÓN DE LOS DISPOSITIVOS .....	50
6.1 Consideraciones previas para la validación .....	50
6.2 Prueba 1: Análisis de propagación con línea de vista .....	53
6.3 Prueba 2: Análisis de propagación en un ambiente controlado .....	57
6.4 Prueba 3: Implementación de la red en condiciones reales.....	61
<b>CONCLUSIONES</b> .....	63
<b>RECOMENDACIONES</b> .....	65
<b>REFERENCIAS BIBLIOGRÁFICAS</b> .....	66
<b>BIBLIOGRAFÍAS</b> .....	67
<b>ANEXOS</b> .....	68

## ÍNDICE DE TABLAS

	Pág.
Tabla 1: Comparación entre las distintas tecnologías de comunicación inalámbrica.	11
Tabla 2: Definición de códigos de funciones públicas.....	19
Tabla 3: Códigos de excepciones.....	20
Tabla 4: Descripción de los pines del radio. ....	27
Tabla 5: Configuración de los pines del microcontrolador. ....	32
Tabla 6: Funciones de la librería del radio con sus entradas y salidas.....	34
Tabla 7: Velocidades disponibles para la transmisión por radio.....	35
Tabla 8 : Funciones MODBUS implementadas para la validación. ....	51
Tabla 9: Porcentaje de paquetes perdidos [%] para la Prueba 1. ....	55
Tabla 10: Error promedio para cada velocidad para la Prueba 1. ....	56
Tabla 11: Porcentaje de error promedio para las distintas velocidades. ....	59
Tabla 12 : Resultados de la implementación en condiciones reales. ....	62
Tabla 13: Tabla de datos de la medición de la RSSI [dBm] promedio para la Prueba 1.....	69
Tabla 14: Tabla de datos con la medición de la RSSI [dBm] promedio para la Prueba 2.....	69
Tabla 15: Tabla de datos con el porcentaje de paquetes perdidos para la Prueba 2. ..	70

## ÍNDICE DE FIGURAS Y/O GRÁFICOS

	Pág.
Figura 1: Topologías tipo (a) bus, (b) árbol, (c) anillo y (d) estrella. ....	7
Figura 2: Modelo de referencia OSI.....	12
Figura 3: PDU del MODBUS en la capa de aplicación.....	17
Figura 4: PDU del MODBUS en la capa de enlace. ....	17
Figura 5: Trama MODBUS RTU.....	19
Figura 6: Red propuesta. ....	22
Figura 7: Diagrama de bloques del hardware. ....	24
Figura 8: Diagrama de pines del (a) RFM69HW y (b) RFM69HCW. ....	26
Figura 9: Máquina de estados del firmware.....	30
Figura 10: Diagrama de flujo simplificado del Estado 0. ....	41
Figura 11: Diagrama de flujo simplificado del Estado 1. ....	43
Figura 12: Diagrama de flujo simplificado del Estado 2. ....	44
Figura 13: MODBUS encapsulado para el envío por el radio. ....	45
Figura 14: Diagrama de flujo simplificado del Estado 3. ....	47
Figura 15: Diagrama de flujo simplificado del Estado 4. ....	49
Figura 16: Captura de pantalla del programa de validación. ....	53
Figura 17: (a) RSSI vs Distancia para la PEv1. (b) RSSI vs Distancia para la PEv2. ....	54
Figura 18: (a) RSSI vs Distancia a 19200bps. (b) RSSI vs Distancia a 9600bps. (c) RSSI vs Distancia a 4800bps. (d) RSSI vs Distancia a 1200bps. ....	56
Figura 19: (a) RSSI vs N° de pisos para la PEv1. (b) RSSI vs N° de pisos para la PEv2.....	58

Figura 20: (a) Porcentaje de paquetes perdidos vs N° de pisos para la PEv1. (b) Porcentaje de paquetes perdidos vs N° de pisos para la PEv2. .... 58

Figura 21: (a) RSSI vs N° de pisos a 19200bps. (b) RSSI vs N° de pisos a 9600bps. (c) RSSI vs N° de pisos a 4800bps. (d) ) RSSI vs N° de pisos a 1200bps. .... 60

Figura 22: Configuración general para la Pueba 3..... 62

## SIGLAS Y ACRÓNIMOS

**ADU:** Application Data Unit (Unidad de datos de aplicación).

**CONATEL:** Comisión Nacional de Telecomunicaciones.

**CRC:** Cyclic Redundancy Check (Comprobación de redundancia cíclica).

**FIFO:** First In First Out (Primero en entrar, primero en salir).

**FSK:** Frequency-Shift Keying (Modulación por desplazamiento en frecuencia).

**ISO:** International Organization of Standardization (Organización internacional de normalización).

**LSB:** Least Significant Bit (Bit menos significativo).

**MAC:** Medium Access Control (Control de acceso a la red).

**MSB:** Most Significant Bit (Bit más significativo).

**NFC:** Near-field communication (Comunicación de campo cercano).

**OOK:** On-off keying (Modulación digital de amplitud).

**OSI:** Open System Interconnection (Modelo abierto de interconexión de sistemas).

**PDU:** Protocol Data Unit (Unidad de datos de protocolo).

**PE:** Puerta de Enlace.

**PEvX:** Puerta de Enlace versión X.

**RF:** Radio Frequency (Radiofrecuencia).

**RSSI:** Received Signal Strength Indicator (Indicador de fuerza de la señal recibida)

**RS232:** Estándar EIA-232 TIA-232.

**RS485:** Estándar EIA-485 TIA-485.

**RTU:** Remote Terminal Unit (Unidad terminal remota).

**UDP:** User Datagram Protocol.

**UHF:** Ultra High Frequency (Ultra alta frecuencia).

**WLAN:** Wireless Local Area Network (Redes inalámbricas de área local).

**WMAN:** Wireless Metropolitan Area Networks (Redes inalámbricas de área metropolitana).

**WPAN:** Wireless Personal Area Network (Redes inalámbricas de área personal).

**WWAN:** Wireless Wide Area Network (Redes inalámbricas de área amplia).

# INTRODUCCIÓN

La tecnología inalámbrica ha estado disponible desde hace décadas, pero no fue sino hasta los últimos años que ha evolucionado, incrementado su confiabilidad y disminuido su costo lo suficiente como para hacer su aparición en el ambiente industrial. Este tipo de tecnologías han permitido la transmisión de información a larga distancia de manera eficiente, lo que abre un mundo de posibilidades a los diseñadores para la implementación de soluciones en el ámbito industrial.

Las redes inalámbricas ofrecen una serie de ventajas como la movilidad, el bajo mantenimiento y la posibilidad de implementarse cubriendo grandes distancias. Resultan particularmente útiles en lugares donde el uso de cables es muy costoso o donde su instalación resulta imposible debido a las condiciones geográficas. Adicionalmente las redes inalámbricas son de fácil instalación y modificación en relación a los cables y otros medios físicos.

Al observar las ventajas de las redes inalámbricas, y tomando en cuenta que el protocolo MODBUS RTU es uno de los protocolos industriales más usados a nivel mundial, resulta clara la necesidad del diseño de una red industrial MODBUS RTU inalámbrica.

En el presente trabajo especial de grado se propone la implementación de una red industrial inalámbrica mediante el diseño de un dispositivo de puerta de enlace que permita interconectar distintos dispositivos MODBUS RTU.

La estructura del trabajo está conformada por capítulos, el primer capítulo presenta la justificación de la realización del trabajo, así como los objetivos planteados para el mismo. El segundo capítulo expone el marco teórico, donde se presenta una breve introducción a las redes de comunicación inalámbrica, una comparación entre las tecnologías inalámbricas más utilizadas y una descripción general del protocolo MODBUS.

En el tercer capítulo, se proponen las características de diseño del dispositivo de puerta de enlace con sus correspondientes consideraciones y limitaciones. La descripción del hardware del equipo se realiza en el cuarto capítulo, donde se describen cada uno de los módulos que conforman el dispositivo de puerta de enlace.

En el quinto capítulo se lleva a cabo una descripción detallada del proceso de diseño del código del programa o firmware del dispositivo, iniciando con una descripción general del funcionamiento del programa, pasando por la construcción de una librería para el manejo del dispositivo inalámbrico y finalmente, describiendo cada uno de los estados que conforman el funcionamiento de la puerta de enlace.

En el sexto capítulo se documentan todos los resultados obtenidos de las pruebas de validación, donde se realizaron los análisis de propagación de los dispositivos de puerta de enlace diseñados y la demostración de su implementación



en un ambiente real. Finalmente se presentan las conclusiones y recomendaciones que derivan de la elaboración del trabajo especial de grado.

# **CAPÍTULO I**

## **MARCO REFERENCIAL**

### **1.1 Justificación e importancia**

En la actualidad, las redes de comunicación MODBUS RTU se emplean en todo tipo de industrias y empresas para cumplir con diversas funciones que varían desde el control y optimización de procesos hasta el monitoreo de variables y domótica. Sin embargo, la implementación de las redes puede presentar una serie de inconvenientes, en particular los altos costos y la dificultad asociada a su instalación y mantenimiento. De esta manera, una posible solución a las situaciones planteadas anteriormente es el diseño de un dispositivo económico que permita implementar una red inalámbrica MODBUS RTU.

En el mercado existe gran variedad de dispositivos que cumplen con la función de implementar redes inalámbricas MODBUS, sin embargo, el alto costo, la falta de servicio técnico y la imposibilidad de importarlos con facilidad dificulta su adquisición. Es por esto que el diseño e implementación de equipos de bajo costo y de fácil uso resulta de gran ayuda en el mercado actual y sirve como apoyo para solucionar los problemas planteados.

## **1.1 Objetivo general**

Diseñar una red inalámbrica de comunicación entre dispositivos industriales empleando el protocolo MODBUS RTU.

## **1.2 Objetivos específicos**

1. Recopilar información sobre el protocolo de comunicación MODBUS RTU.
2. Recopilar información sobre las diferentes tecnologías inalámbricas disponibles y seleccionar la más adecuada para el dispositivo.
3. Especificar la topología de la red a implementar.
4. Diseñar el firmware del dispositivo de puerta de enlace para que establezca la comunicación inalámbrica entre dispositivos MODBUS RTU.
5. Implementar la red inalámbrica MODBUS RTU mediante el uso de las puertas de enlace diseñadas.
6. Validar los dispositivos y la red diseñada realizando pruebas de campo.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1 Redes de comunicación**

Una red de comunicación es un conjunto de nodos conectados entre sí capaces de intercambiar información con el objeto de transmitir un mensaje. La conexión entre los nodos puede ser de dos tipos: punto a punto y multipunto. Una conexión punto a punto se refiere a dos nodos conectados directamente entre sí, mientras que una conexión multipunto ocurre cuando un nodo está conectado a varios nodos al mismo tiempo. Entre los elementos más importantes de una red de comunicación se encuentran la topología de la red, el medio de transmisión y el modo de acceso a la red [1].

La topología de la red se refiere a como los nodos de la misma están interconectados. Las topologías más comunes para las redes son la topología tipo bus, tipo árbol, tipo anillo y tipo estrella, mostradas en la Figura 1. A continuación se describe brevemente cada una de ellas:

- Topología tipo bus: en esta topología todos los nodos de la red están conectados por el mismo medio de manera lineal en una conexión de tipo multipunto. Cuando un nodo inicia la comunicación el mensaje se propaga por todo el medio y es recibido por el resto de los nodos. Se suelen usar

terminadores en los extremos del bus para absorber las señales y eliminarlas del bus.

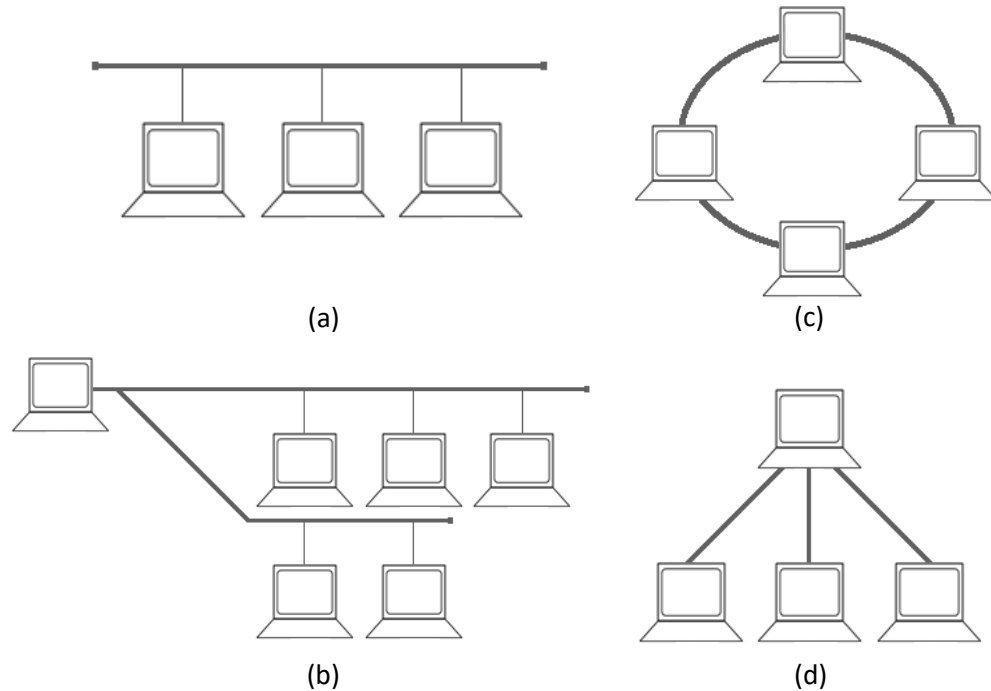


Figura 1: Topologías tipo (a) bus, (b) árbol, (c) anillo y (d) estrella.

- Topología de tipo árbol: es un caso generalizado de la topología tipo bus en la que ocurren una o más bifurcaciones del bus inicial, estas bifurcaciones son llamadas ramas. De las ramas pueden dividirse otras ramas, creando configuraciones más complejas, sin embargo el ciclo no es cerrado en ningún momento. De la misma manera que en la topología de tipo bus, los mensajes transmitidos se propagan por toda la red y son recibidos por todos los nodos.
- Topología tipo anillo: está conformada por una serie de repetidores conectados punto a punto entre sí, formando un ciclo cerrado. Cuando uno de los nodos envía un mensaje, este recorre el anillo en una dirección hasta regresar a la fuente, donde se remueve del sistema. Ya que los nodos comparten

la red, es necesario que exista algún tipo de control en el medio para evitar que dos nodos intenten comunicarse al mismo tiempo.

- Topología tipo estrella: todos los nodos están conectados a un nodo principal mediante una conexión punto a punto. Normalmente el nodo principal tiene dos formas de operar:
  - Modo de broadcast o emisión: el nodo principal envía un mensaje a todos los nodos de la red y esta actúa virtualmente como un bus de datos.
  - Modo de interruptor: el nodo principal actúa como un interruptor maestro que dirige la trama directamente al nodo destino, solo él recibe el mensaje y solo él responde al nodo principal.

Para que los nodos puedan establecer comunicación entre sí, deben poder acceder a un medio de transmisión. En lo sucesivo de este trabajo especial de grado, todas las formas de comunicación se referirán específicamente al uso de señales electromagnéticas, que pueden ser transmitidas a través de un medio guiado o no guiado. Los medios guiados de transmisión más comunes son el par trenzado, el cable coaxial y la fibra óptica, mientras que los medios no guiados se refieren a transmisión inalámbrica, que puede ser por satélite, emisiones de radio, entre otras. [1]

Dependiendo del medio de transmisión y de la topología de la red es necesaria la presencia de un elemento que controle el acceso a la red, de allí surge el Medium Access Control o control de acceso a la red. El MAC es el encargado de darle autorización a los nodos para acceder al medio, puede ser centralizado, cuando es una entidad que autoriza a los nodos a transmitir información o descentralizado cuando son todos los elementos de la red los que trabajan de manera dinámica para establecer el orden de acceso al medio [1].

## 2.2 Redes de comunicación inalámbrica

Una red inalámbrica se basa en una red de comunicación que no depende de un medio físico guiado, sino de un medio de transmisión no guiado. El medio de transmisión inalámbrico más común en las telecomunicaciones son las transmisiones por radio frecuencia.

Las redes de comunicación inalámbrica pueden ser clasificadas de acuerdo a las distancias que son capaces de cubrir:

- Redes WWAN: Son las redes inalámbricas de área amplia, capaces de transmitir a muy largas distancias que pueden cubrir regiones geográficas completas.
- Redes WMAN: Las redes inalámbricas de área metropolitana son capaces de cubrir ciudades completas, utilizadas comúnmente para interconectar redes WLAN.
- Redes WLAN: Las redes inalámbricas de área local se caracterizan por ser redes de corto alcance que abarcan una zona limitada. Son las redes más utilizadas, normalmente privadas y utilizadas para conectar equipos finales.
- Redes WPAN: Las redes de área personal se caracterizan por su rango muy limitado, normalmente de pocos metros.

Existen diferentes tecnologías asociadas con comunicación por radio, cada una de ellas trabaja en una banda de frecuencias y cuenta con características y aplicaciones diferentes. En las telecomunicaciones, las bandas de frecuencia están comprendidas desde los 3KHz hasta los 300GHz. En Venezuela la regulación del

espectro radioeléctrico está encargada a la Comisión Nacional de Telecomunicaciones (CONATEL).

Entre las tecnologías inalámbricas de transmisión de datos más comunes se encuentran:

- Wi-Fi: es una tecnología de transmisión inalámbrica que permite establecer redes locales inalámbricas basadas en el estándar IEEE 802.11. Este estándar surgió bajo la necesidad de garantizar la interconectividad de los dispositivos inalámbricos. Trabaja en las bandas de 2.4GHz y 5.8GHz bajo el estándar IEEE 802.11, con un alcance de hasta 100m. Las velocidades de transmisión pueden llegar hasta los 600Mbps dependiendo de las especificaciones utilizadas. Es uno de los sistemas de WLAN más utilizados en la actualidad y su principal función es la sustitución de redes LAN de alta velocidad.
- Bluetooth: es un estándar de transmisión de datos WPAN que trabaja en la banda de 2.4GHz. Tiene un alcance promedio de 10m y dependiendo de su versión de implementación puede transmitir velocidades comprendidas entre los 25Mbps hasta los 50Mbps. Es utilizado ampliamente para la conexión de dispositivos portátiles de uso personal
- Transceptores en bandas libres: se refiere a todos los dispositivos transceptores de radio que utilizan las bandas libres de licencia (433MHz, 868MHz y 915MHz) para establecer redes de comunicación. El alcance y velocidad depende de las características de cada dispositivo.
- NFC: es una tecnología de comunicación punto a punto de corta distancia, capaz de transmitir en la banda de 13.52MHz. Trabaja con el estándar ISO 13157 y puede llegar a transmitir velocidades de hasta 424Kbps.



En la Tabla 1 se comparan los distintos tipos de tecnologías inalámbricas con algunas de sus características más importantes para las consideraciones de este proyecto.

Característica	Wi-Fi	Bluetooth	Transceptores de banda libre	NFC
Banda de operación	2,4GHz 5,8GHz	2,4GHz	433MHz 868MHz 915MHz	13,52MHz
Tipo de red	WLAN	PLAN	WLAN WPAN	Punto a Punto
Consumo de potencia	Moderado	Bajo	Bajo	Bajo
Rango	<100m	<10m	Depende del dispositivo	<10cm
Velocidad	Hasta 1300mbps	25Mbps 50Mbps	Depende del dispositivo	424Kbps

Tabla 1: Comparación entre las distintas tecnologías de comunicación inalámbrica.

### 2.3 Modelo de referencia OSI

Debido a la urgente necesidad por normalizar las redes de comunicación entre computadores, en 1984, ISO creó un modelo que caracteriza y estandariza las funciones internas de los sistemas interconectados [2]. El modelo abierto de interconexión de sistemas OSI permite solventar los problemas de incompatibilidad de los distintos sistemas, estableciendo un conjunto de normas aplicables a todas las redes.

El modelo de referencia OSI, describe como la información viaja de una aplicación informática en un computador a otra aplicación en otro computador mediante una red. Es un modelo compuesto por siete capas, mostradas en la Figura 2, cada una especificando una función particular de la red de comunicación [3].

El modelo OSI establece una estructura de soporte o guía para la comunicación entre computadoras, pero el modelo en sí mismo no es un método de comunicación. La comunicación se lleva a cabo mediante el uso de un protocolo de comunicación. Un protocolo de comunicación se refiere a todas las reglas y convenciones que rigen el intercambio de información en una red. Un protocolo implementa la función de una o varias capas del modelo OSI. [3]

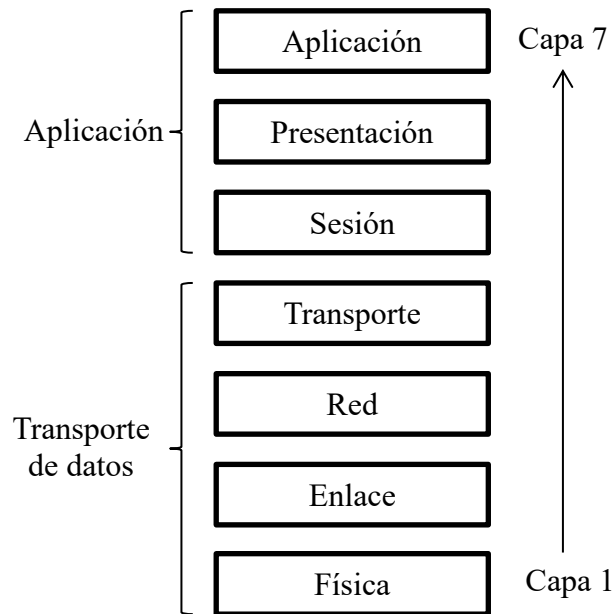


Figura 2: Modelo de referencia OSI.

Los procesos que contribuyen al movimiento de información en la red son divididos en tareas o grupos más pequeños, que son asignados a una de las siete capas del modelo OSI. Las capas son relativamente auto contenidas, y pueden ser implementadas de manera independiente, lo que permite la modificación de una de las capas sin que las demás sean afectadas [3].

Las tres capas superiores del modelo OSI están asociadas a la aplicación, y suelen ser implementadas en software, mientras que las cuatro capas inferiores se encargan del transporte de datos en la red. La capa física y la capa de enlace se implementan tanto en software como en hardware. Las capas restantes asociadas al transporte de datos se pueden implementar tanto en software como en hardware. [2][3]

Las distintas capas del modelo OSI se definen de la siguiente manera:

### 2.3.1 Capa física

La capa física define las especificaciones eléctricas y físicas de la conexión de datos. Como por ejemplo, los niveles de voltaje, distancias máximas de transmisión, conexiones físicas, velocidades de transmisión de datos, entre otros [3].

Es en la capa física donde se establece el medio de comunicación y en donde se transforman los bits de datos provenientes de la capa de enlace en una señal adecuada para la transmisión en un medio físico. Adicionalmente, debido a que es en

la capa física donde se realiza la conexión de datos, es en ella en donde se establece la topología de la red.

### 2.3.2 Capa de enlace

El propósito de la capa de enlace es el de proporcionar los medios y procedimientos para establecer, mantener y liberar los enlaces de datos entre las entidades de la red [2]. Es responsable de identificar y encapsular los protocolos de la red, así como controlar la detección de errores producidos en la capa física y la sincronización de la trama.

### 2.3.3 Capa de red

La capa de red se encarga de las funciones de direccionamiento de la red, lo que permite combinar múltiples enlaces de datos para formar una red interconectada [3]. Esto se logra mediante el direccionamiento lógico de los dispositivos.

### 2.3.4 Capa de transporte

La capa de transporte proporciona una serie de servicios confiables de transporte de datos para la red interconectada que son transparentes para las capas superiores. Entre otras funciones, está encargada de realizar el control de flujo para que un dispositivo no pueda recibir más datos de los que puede procesar y permitir el

multiplexado de datos provenientes de distintas aplicaciones para ser enviados por el mismo enlace de datos. [3]

### 2.3.5 Capa de sesión

El propósito de la capa de sesión es el de proveer de servicios que asistan en la interacción entre entidades de la red. Esto se logra estableciendo la conexión y desconexión de entidades mediante servicios de administración de sesión, y realizando el control, intercambio y sincronización de datos entre dos entidades mediante servicios de dialogo de sesión. [2]

### 2.3.6 Capa de presentación

La capa de presentación se encarga de establecer los servicios que pueden ser seleccionados por la capa de aplicación para interpretar el significado de las estructuras de datos abstractas intercambiadas por la red. Los servicios de presentación definidos deben ser independientes de la locación, permitiendo así que la información enviada por la capa de aplicación de un sistema pueda ser comprendida por la capa de aplicación de otro sistema. [2][3]

### 2.3.7 Capa de aplicación:

Es la capa más cercana al usuario, pero no interactúa directamente con él. La comunicación se realiza mediante el programa o aplicaciones de comunicación, por lo

que tanto la capa de aplicación como el usuario interactúan directamente con la aplicación de software. La capa de aplicación identifica a los involucrados en la comunicación, determina la disponibilidad de recursos de la red y realiza la sincronización de la comunicación. [3]

## **2.4 El protocolo MODBUS**

MODBUS es un protocolo de comunicaciones originalmente ubicado en la capa de aplicación del modelo OSI, que ofrece comunicación cliente/servidor entre distintos dispositivos y medios. Ha sido el protocolo preferido en la industria desde 1979 [4]. “El protocolo MODBUS se ha expandido para incluir implementaciones a través de protocolo serial, TCP/IP y el User Datagram Protocol (UDP)” [5]:

A continuación se describen las características más importantes del protocolo MODBUS sobre una línea serial, que se encuentra ubicado en la segunda capa del modelo OSI. [6]

### **2.4.1 MODBUS en la capa de enlace**

El protocolo MODBUS sobre una línea serial es un protocolo jerárquico de tipo maestro/esclavo, en el que uno de los nodos, el maestro, envía solicitudes explícitas a los nodos esclavos para que sean procesadas y respondidas. Los nodos esclavos no pueden comunicarse entre sí, y no transmiten datos sin la previa solicitud del maestro. El maestro solo puede iniciar un intercambio de información a la vez.

Cada uno de los esclavos es identificado con una dirección, que debe ser única para cada bus de datos. Estas direcciones pueden ir desde la 1 hasta la 247. Las direcciones comprendidas entre la 248 y la 255 están reservadas por el protocolo, mientras que la dirección 0 se encuentra designada para el modo de broadcast. El maestro no posee una dirección específica.

En la capa de aplicación la trama MODBUS está conformada por una unidad de datos de protocolo (PDU), mostrada en la Figura 3, que contiene la función que desea implementar el maestro y, de haberlos, los datos asociados a ella. Al insertar el protocolo MODBUS en la capa de enlace se le asocian dos nuevos elementos a la PDU: La dirección del esclavo y la comprobación de redundancia cíclica (CRC), como se muestra en la Figura 4.

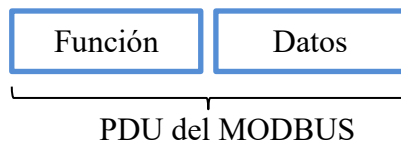


Figura 3: PDU del MODBUS en la capa de aplicación.

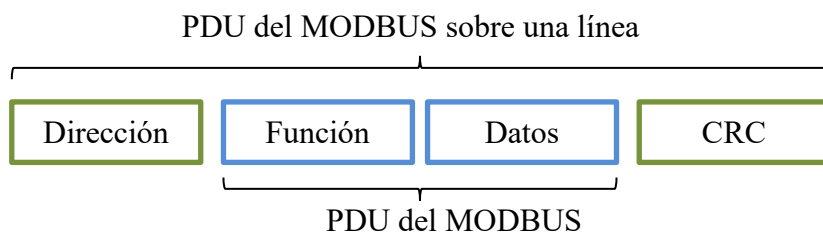


Figura 4: PDU del MODBUS en la capa de enlace.

El maestro tiene dos maneras de hacer solicitudes a los esclavos: el modo unicast y el modo broadcast [3]. En el modo unicast el maestro envía una petición a un único esclavo y este procesa y responde la petición, mientras que en el modo de broadcast el maestro envía un mensaje a todos los esclavos empleando la dirección 0, los esclavos procesan la petición pero no emiten una respuesta.

De acuerdo al contenido de los bits de la PDU el protocolo MODBUS sobre una línea serial presenta dos modos de transmisión: el modo RTU y el modo ASCII [6]. Este trabajo especial de grado se enfocará en el MODBUS RTU.

#### 2.4.1.1 MODBUS RTU

En el MODBUS RTU cada byte enviado está conformado por dos caracteres hexadecimales de 4 bits. Para enviar un byte se emplea 1 bit de inicio, los 8 bits que conforman el byte a enviar, 1 bit de paridad y 1 bit de parada. Los bytes de datos son enviados del bit menos significativo (LSB) al bit más significativo (MSB). [6]

La trama de MODBUS RTU sigue con el esquema básico de la PDU de MODBUS sobre una línea serial, y está limitada a una longitud máxima de 256 bytes. En el MODBUS RTU el sistema determina que se finalizó el envío de una trama mediante un tiempo de silencio en la red. Este tiempo queda especificado de acuerdo a la velocidad de transmisión como 3.5 veces el tiempo de envío de un byte por la red. Para velocidades sobre los 19200Bd el protocolo recomienda una espera mínima de 2ms [5]. En la Figura 5 se presenta un esquema con la PDU del MODBUS RTU y los periodos de espera necesarios.



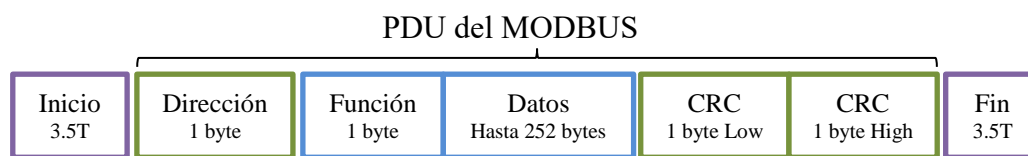


Figura 5: Trama MODBUS RTU.

Las distintas funciones que se pueden operar en una red MODBUS pueden ser funciones de uso público, definidas por el usuario y funciones reservadas. Las funciones de uso público están definidas en el protocolo y son únicas, pero no todos los dispositivos MODBUS tienen habilitadas todas las funciones públicas. [4]. En la Tabla 2 se encuentran las funciones de uso público definidas para el protocolo MODBUS RTU.

		Función	Código
Acceso a Datos	Acceso a Bits	Leer entrada discreta	02
		Leer bobinas	01
		Escribir una bobina	05
		Escribir múltiples bobinas	15
	Acceso de 16 Bits	Leer registro de entrada	04
		Leer múltiples registros	03
		Escribir un registro	06
		Escribir múltiples registros	16
		Leer/Escribir múltiples registros	23
		Enmascarar escritura de registro	22
		Leer FIFO	24
	Acceso a Archivos	Leer archivo	20
		Escribir archivo	21
	Diagnóstico	Leer estatus de excepción	07
Diagnostico		08	
Obtener contador de eventos		11	
Obtener log de eventos		12	
Reportar ID		17	
Leer identificación de dispositivo		43	

Tabla 2: Definición de códigos de funciones públicas.

Los esclavos emplean los códigos de excepción para indicar al maestro que ha ocurrido un error, que puede deber a varias razones, como por ejemplo el recibir una solicitud incompleta, o que el esclavo no tenga implementado el código de la función solicitada. En la Tabla 3 se presentan los códigos de excepciones más utilizados. [5]

Código de excepción	Descripción
01	El código de la función recibida no está habilitado
02	La solicitud intento acceder a una dirección no válida
03	La solicitud tenía datos incorrectos
04	Ocurrió un error irrecuperable al procesar la solicitud

Tabla 3: Códigos de excepciones.

#### 2.4.2 MODBUS en la capa física

A nivel de la capa física el protocolo MODBUS sobre una línea serial puede ser implementado por distintas interfaces, la más común es el RS485 de par trenzado, aunque también está disponible para el estándar RS485 de cuatro cables y en el RS232 para conexiones cortas de tipo punto a punto [6].

En el estándar MODBUS de par trenzado todos los dispositivos están conectados en paralelo por tres cables: el común y el bus de datos conformado por el par trenzado. La velocidad para la transmisión de datos suele ser de 9600Bd y los dispositivos MODBUS también deben implementar la velocidad de 19200Bd. [6]

## **CAPÍTULO III**

### **PROPUESTA Y REQUERIMIENTOS DEL SISTEMA**

#### **3.1 Propuesta del sistema**

El dispositivo a desarrollar consiste en un módulo de puerta de enlace (PE) que permite establecer una red inalámbrica de comunicaciones utilizando el protocolo MODBUS RTU. Se consideró inicialmente que la puerta de enlace sea empleada en zonas industriales donde se desee establecer comunicación inalámbrica entre terminales remotas y el maestro. El dispositivo se comunicará tanto con las terminales remotas como con el maestro por medio de una interfaz serial RS485 empleando el protocolo MODBUS RTU. La conexión mediante la interfaz RS485 podrá ser independiente de si se conecta directamente a las terminales remotas o a parte de la red, por lo cual la puerta de enlace puede ser anexada posteriormente para ampliar la red preexistente o agregar otras secciones a la misma. En la Figura 6 se presenta un diagrama general de una red preexistente a la que se le agrega un nuevo tramo empleando las puertas de enlace diseñadas. Se pretende que el uso del dispositivo sea completamente transparente para el usuario final.

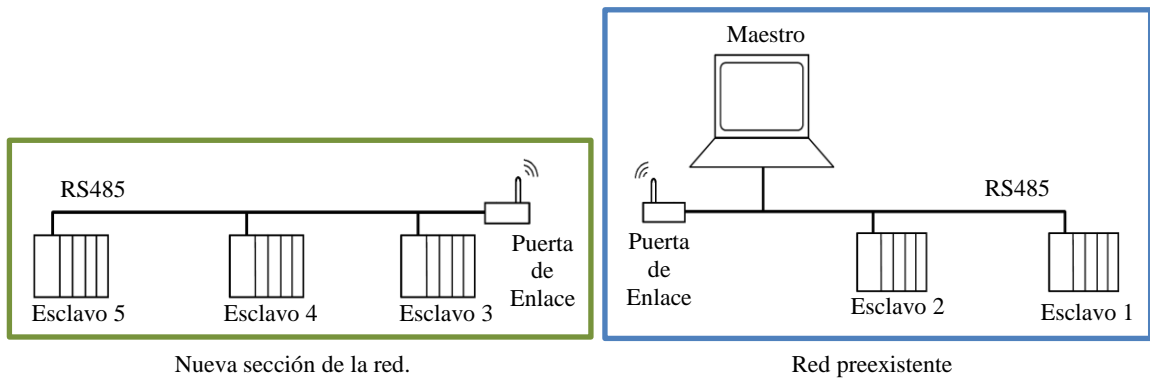


Figura 6: Red propuesta.

### 3.2 Requerimientos del dispositivo

Para que el dispositivo cumpla con el objetivo de establecer una red inalámbrica MODBUS RTU debe cumplir con los siguientes requisitos:

- El dispositivo se debe comunicar mediante la interfaz serial RS485 ya sea directamente a un dispositivo MODBUS RTU o a parte de una red MODBUS RTU preexistente.
- De acuerdo al estándar MODBUS RTU el dispositivo debe poder establecer comunicación a 19200Bd y a 9600Bd, siendo 19200Bd la velocidad predeterminada.
- La transmisión de caracteres por serial se llevará a cabo cumpliendo con las siguientes características: un bit de inicio, 8 bits de datos, bit de paridad (par o ninguna) y un bit de parada.
- El tiempo entre tramas para ambas velocidades será de mínimo 3ms.
- Las tramas deben alcanzar a todos los dispositivos de la red, en particular para permitir al maestro comunicarse con todos los esclavos mediante el modo de broadcast.

- Los dispositivos diseñados deben ser capaces de comunicarse entre ellos de manera inalámbrica mientras se mantenga la línea de vista, las distancias a las que se realiza la comunicación serán evaluadas durante el diseño.
- Las tramas recibidas por el dispositivo mediante el RS485 deben ser enviadas a los otros dispositivos de manera inalámbrica, y las tramas recibidas de manera inalámbrica deben ser enviadas a la red mediante la interfaz RS485.
- Se busca garantizar la seguridad de la información transportada por la red, por lo que se tomarán en cuenta elementos como la encriptación de la información.

## CAPÍTULO IV

### DESCRIPCIÓN DEL HARDWARE

#### 4.1 Descripción general del hardware

El hardware del dispositivo está dividido en cuatro módulos: el módulo de comunicación inalámbrica, el módulo de control, la interfaz RS485 y el módulo de alimentación.

El montaje del dispositivo se llevó a cabo mediante la utilización de una placa prediseñada en la que se colocaron los componentes seleccionados para cada módulo. En la Figura 7 se presenta un diagrama de bloques del hardware. A continuación se procede a explicar en detalle cada uno de los módulos.

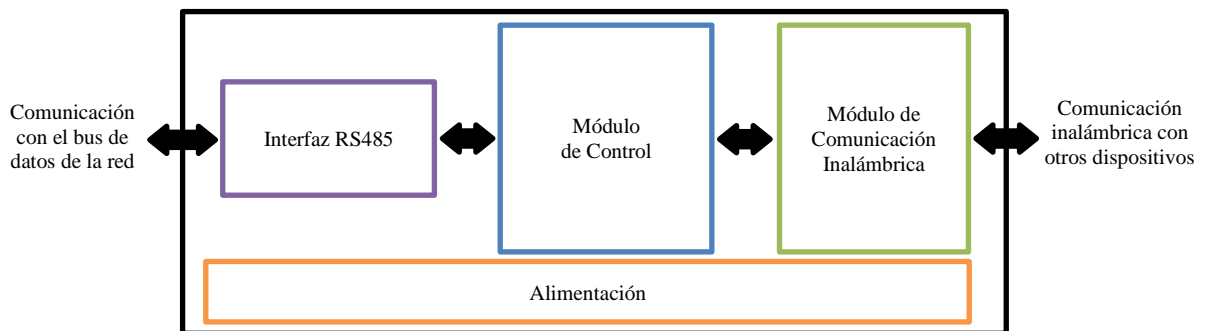


Figura 7: Diagrama de bloques del hardware.

## 4.2 El módulo de comunicación inalámbrica

La función del módulo de comunicación inalámbrica es proveer al módulo de control de un servicio que le permita comunicarse con los otros dispositivos de manera inalámbrica. Tomando en cuenta las tecnologías investigadas en el Capítulo II se seleccionaron los transceptores RFM69HW y RFM69HCW de HoperRF Electronic, que utiliza la banda de UHF de 433MHz.

La única diferencia entre los modelos RFM69HW y RFM69HCW es el tamaño del dispositivo y el mapeado de los pines: el RFM69HW mide 19mmx16mm, mientras que el RFM69HCW mide 16mmx16mm. Se plantea la utilización del modelo RFM69HW para realizar prototipos (PEv1) que determinen la factibilidad del proyecto y permitan hacer las pruebas iniciales, mientras que la versión RFM69HCW será utilizada para implementar la versión final de las puertas de enlace diseñadas (PEv2). Se utilizaron dos placas prediseñadas distintas, una para la PEv1 y otra para la PEv2.

En el montaje de la PEv1 se empleó una antena de tipo mono polo de un cuarto de longitud de onda, que para los 433MHz es de 164.5mm. Para el montaje de la PEv2 se utilizó un cable coaxial de  $50\Omega$  conectado a un puerto SMA hembra, donde se colocó una antena comercial para la banda de 433MHz.

El RFM69HW/HCW es un transceptor de baja potencia que emplea la tecnología del chip SX1231/H de Semtech para transmitir en la banda de 433MHz [7][8]. Es ideal para aplicaciones de radio de bajo consumo, tiene una sensibilidad de hasta -120dBm, una potencia de salida entre los -18dBm y 20dBm y es capaz de

modular en FSK y OOK, con la ventaja de poder seleccionar los parámetros de la modulación por software.

El radio se alimenta de 3.3v, garantizando su correcto funcionamiento para un voltaje de alimentación entre los 2.4v y 3.6v y un consumo máximo de 130mA durante la transmisión para la configuración de 20dBm de potencia [8]. El rango de temperaturas en el que funciona de manera adecuada está comprendido entre los -20°C y los 70°C.

La modificación de los registros de configuración y el manejo de paquetes del radio se lleva a cabo mediante el uso de una interfaz SPI incluida. El radio adicionalmente cuenta con una serie de pines reprogramables con distintas funciones, entre ellas la interrupción de paquete recibido e interrupción de paquete enviado. En la Figura 8 se muestra el diagrama de pines para ambas versiones del RFM69HW/HCW y en la Tabla 4 se describe el uso de cada uno de ellos.

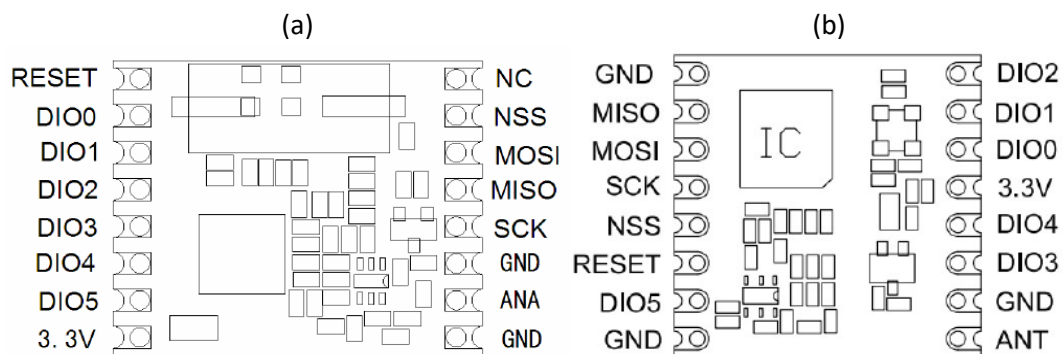


Figura 8: Diagrama de pines del (a) RFM69HW y (b) RFM69HCW.



Nombre	Tipo	Descripción
Reset	I	Interruptor de reset.
DIO0	I/O	Pin reprogramable del radio.
DIO1	I/O	Pin reprogramable del radio.
DIO2	I/O	Pin reprogramable del radio.
DIO3	I/O	Pin reprogramable del radio.
DIO4	I/O	Pin reprogramable del radio.
DIO5	I/O	Pin reprogramable del radio.
3,3V	-	Alimentación
ANA	-	Conexión de la antena (Salida/Entrada RF)
GND	-	Tierra
SCK	I	Entrada del reloj del SPI
MISO	O	Salida de datos del SPI
MOSI	I	Entrada de datos del SPI
NSS	I	Enable del SPI
NC	-	Conectado a tierra

Tabla 4: Descripción de los pines del radio.

### 4.3 Módulo de control

El módulo de control está conformado por un microcontrolador que se encarga de realizar la comunicación entre los distintos módulos, en recibir, procesar y enviar las tramas MODBUS por el RS485 y en enviar y recibir los paquetes al módulo de comunicación inalámbrica.

Gracias a su bajo costo, gran flexibilidad y confiabilidad se seleccionó al PIC24FJ64GA002 como microcontrolador para desarrollar el dispositivo. Este microcontrolador de la empresa Microchip fue programado mediante el uso del PICKit3. Entre las características más relevantes con las que cuenta este microcontrolador se encuentran:

- Oscilador interno de 8MHz.
- 4x PLL que pueden elevar la velocidad del oscilador hasta 32MHz.
- Rango de operación de 2.6v hasta 3.6v.
- 16 pines reprogramables.
- 2 módulos de SPI.
- 2 módulos de UART con soporte para RS485.
- 64K de memoria de programa y 8K de SRAM.

#### **4.4 Interfaz RS485**

Para cumplir con los estándares del protocolo MODBUS RTU la puerta de enlace diseñada debe implementar una interfaz acorde con el estándar RS485 [6]. La comunicación se realiza mediante un bus de dos hilos. La placa prediseñada incluye un MAX487 que cumple con los requisitos establecidos: transceptor RS485-TTL con comunicación half dúplex (que no se puede enviar y recibir al mismo tiempo) controlada con un pin de habilitación, una velocidad máxima de hasta 250kbps y una alimentación de 5v.

#### **4.5 Fuente de alimentación**

La placa prediseñada se alimenta con 5v mediante un regulador externo al sistema, el regulador de voltaje (AP1115BY33) incluido, reduce la tensión a 3.3v requeridos por el microcontrolador y por el radio.

## **CAPÍTULO V**

### **DISEÑO DEL FIRMWARE**

#### **5.1 Descripción general del firmware**

El firmware del dispositivo se diseñó mediante el uso de programa MPLAB X IDE v3.16 de Microchip utilizando el compilador XC16 y el programador PicKit3 para programar el microcontrolador.

El firmware está diseñado en base a una máquina de estados donde cada uno de los estados tiene asociada una función particular. Esto permite la independencia de cada una de las funciones del dispositivo, además de simplificar el diseño del código al separar las distintas funciones en bloques. En la Figura 9 se muestra el diagrama general de estados de la puerta de enlace.

Al encender el dispositivo, este configura los valores iniciales del microcontrolador y envía los registros de configuración al radio por medio del SPI. Configurados el microcontrolador y el radio la máquina entra en el Estado 0, en este estado la máquina aguarda a que se reciba un mensaje, ya sea una trama desde el RS485 o un paquete desde el radio.

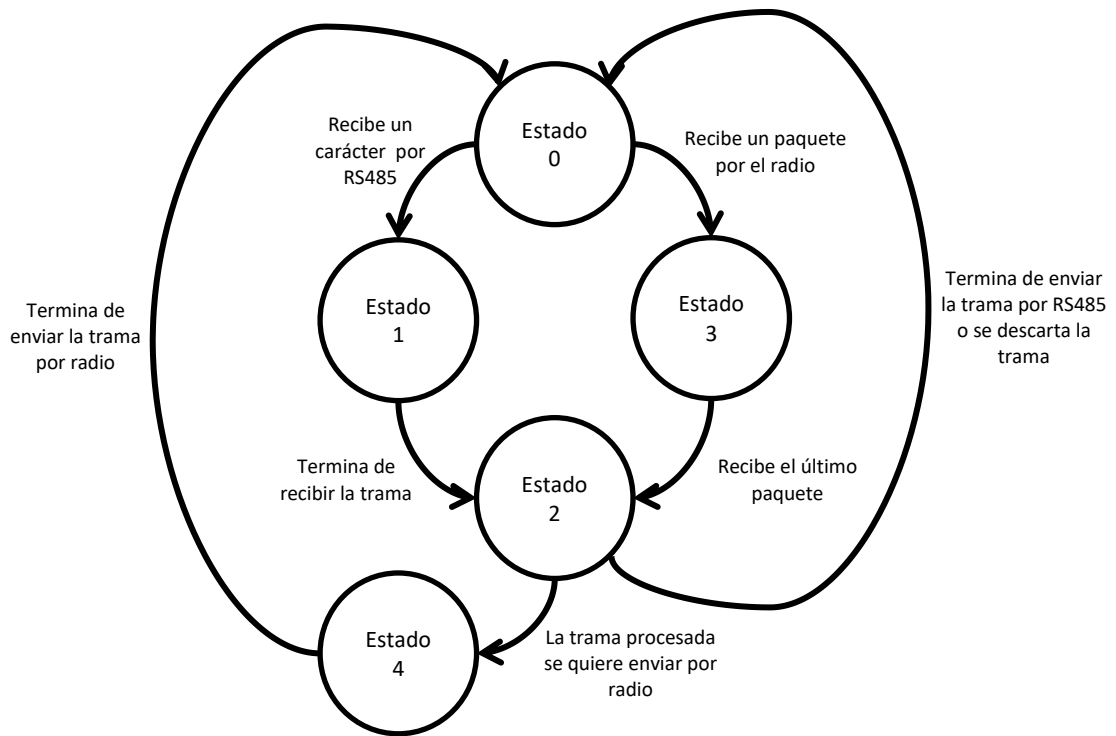


Figura 9: Máquina de estados del firmware.

En caso de que se empiece a recibir una trama desde el RS485 la máquina pasa al Estado 1, en donde recibe uno a uno los caracteres que conforman la trama MODBUS hasta que ocurre un silencio de 3ms, indicando que se recibió toda la trama y enviando al dispositivo al Estado 2.

Si se sale del estado Espera debido a la llegada de un paquete por el radio, la máquina pasa al Estado 3, donde permanece hasta recibir todos los paquetes que conforman la trama. Una vez recibido el último paquete se une la trama y se pasa al Estado 2.

En el Estado 2 se analiza la trama recibida y se decide si se descarta, se reenvía o se responde. En caso de que la trama se descarte la máquina regresa al Estado 0, donde aguarda por una nueva trama. Si la respuesta o el reenvío de la trama debe ser por medio del radio, entonces la máquina pasa al Estado 4, donde la trama se separa en paquetes y se envían uno a uno hasta enviar todo el mensaje, para posteriormente regresar al Estado 0. Finalmente si la respuesta o el reenvío de la trama debe ser mediante la interfaz RS485 entonces el envío se hace de manera inmediata y la máquina permanece en el Estado 2 hasta que envíe la totalidad de la trama para después regresar al Estado 0.

A continuación se procede a describir detalladamente cada uno de los estados.

## **5.2 Configuración inicial del microcontrolador**

Al encender el dispositivo lo primero que se lleva a cabo es la configuración inicial del microcontrolador, en particular la configuración de pines, la configuración del UART para la conexión con la interfaz RS485 y del SPI para la comunicación con el radio, y el manejo de las interrupciones.

Debido a la existencia de las dos versiones del hardware utilizadas se realizaron dos configuraciones de pines independientes para cada una de ellas. En particular en el caso de la PEv1, en el que el radio era muy grande para encajar directamente sobre la tarjeta, por lo cual fue necesario realizar la configuración de manera que la comunicación se llevara a cabo por el Puerto 2 de la tarjeta prediseñada.

Los pines del SPI1 y del UART2 fueron mapeados para ser manejados de manera automática por el microcontrolador. En la Tabla 5 se presentan todas las configuraciones de pines realizadas para cada versión del hardware así como el tipo de configuración.

Función	Nombre del pin en la PEv1 [N° pin]	Nombre del pin en la PEv2 [N° pin]	Tipo de Configuración
Indicador LED	RA1 [3]	RA2 [9]	Salida
RS485 Enable	RA2 [9]	RA4 [12]	Salida
UART2 TX	RB0 [4]	RB11 [22]	Salida
UART2 RX	RB1 [5]	RB13 [24]	Entrada
Reloj SPI1	RB11 [22]	RB9 [18]	Salida
SPI1 SS	RB12 [23]	RB10 [21]	Salida
SPI1 MOSI	RB13 [24]	RB8 [17]	Salida
SPI1 MISO	RB14 [25]	RB7 [16]	Entrada
Interrupción DIO0	RB10 [21]	RB12 [23]	Entrada

Tabla 5: Configuración de los pines del microcontrolador.

El UART2 del microcontrolador se configuró para operar a una tasa de baudios de 19200Bd sin paridad, dejando habilitadas las opciones para trabajar a 9600Bd y con paridad par.

El SPI1 se configuró para trabajar como maestro a 8MHz utilizando el oscilador interno del microcontrolador. Para facilitar el manejo de la data y permitir la comunicación con el radio el modo de comunicación se estableció en 16 bits con el muestreo de la data a la mitad del periodo.

En el caso de las interrupciones se habilitó la interrupción de recepción serial del UART2, la interrupción del Timer 1 para establecer los periodos de silencio del RS485 a 3ms, la interrupción del Timer 2 para controlar el tiempo máximo de envío y recepción de paquetes, y la interrupción externa 1 para el DIO0 del radio, que indica cuando se envía o se recibe un paquete.

### **5.3 Configuración inicial del radio**

Una vez configurado el microcontrolador, este procede a configurar los registros del radio para su correcto funcionamiento. Para llevar esto a cabo, y para facilitar el posterior manejo del radio, se diseñó una librería genérica en C para PIC que permite controlar al radio RFM69HW de manera simple mediante algunas funciones básicas. En la Tabla 6 se presentan todas las funciones que conforman la librería del radio junto con sus respectivas entradas y salidas. A continuación se procede a describir cada una de las rutinas que conforman la librería:

#### **5.3.1 Función RFM69HW\_Config()**

La configuración inicial del radio se lleva a cabo mediante la construcción del vector RF\_Config[], que contiene los registros a modificar y su respectivo valor. La función RFM69HW\_Config() se encarga de enviar al radio, por medio del SPI del microcontrolador, cada uno de los elementos que conforman al vector, y posteriormente pasar al radio al modo de “Standby”, dejándolo disponible para su utilización.

Nombre	Entradas	Salidas	Descripción
RFM69HW_Config()	Ninguna	Ninguna	Configuración inicial del radio.
RF_ReConfig()	Bitrate, PayloadLength y PowerLevel.	Ninguna	Reconfigura el radio con los valores introducidos.
RF_Mode_ChangeTo()	Modo deseado	Ninguna	Cambia el modo al modo deseado.
RF_Send()	Dirección del paquete a enviar.	Ninguna	Envía el paquete por el radio.
RF_PacketRecived()	Ninguna	Ninguna	Guarda el paquete recibido en RF_Packet[.].
RF_ReadRssi()	Ninguna	Ninguna	Mide y guarda el valor de la última RSSI en RSSI.
RF_Encrypt()	Dirección de la clave/Ninguna	Ninguna	Activa el encriptado del radio con la clave suministrada.
RF_Debug()	Ninguna	Ninguna	Lee y envía todos los registros del radio por serial.

Tabla 6: Funciones de la librería del radio con sus entradas y salidas.

Dependiendo de la velocidad de transmisión utilizada fue necesario seleccionar los valores adecuados de desviación de frecuencia ( $F_{dev}$ ) y el ancho de banda del filtro de entrada del receptor ( $RxBW$ ). Dos reglas deben cumplirse para la selección de los parámetros [7]: el índice de modulación debe ser mayor a 0.5 y menor a 10 (Fórmula 1), y el ancho de banda del receptor debe ser menor al doble de la desviación en frecuencia (Fórmula 2).

$$0.5 \leq \beta = \frac{2 F_{dev}}{Bitrate} \leq 10 \quad (1)$$

$$RxBW > 2 F_{dev} \quad (2)$$



La velocidad por defecto seleccionada fue de 19200bps, pero el radio puede ser configurado posteriormente con alguna de las velocidades presentadas en la Tabla 7, donde también se presentan los valores de desviación en frecuencia y de ancho de banda seleccionados para cada caso.

Velocidad [bps]	Fdev [K Hz]	RxBw [K Hz]
19200	20	62,5
9600	15	25
4800	6	12,5
1200	5	10,4

Tabla 7: Velocidades disponibles para la transmisión por radio.

La frecuencia de la onda portadora fue fijada en 433MHz, y se seleccionó el modo de operación por paquetes del radio, con un paquete de 64bytes, palabra de sincronización de 3 bytes, preamble (preámbulo) de 1 byte, se activó el CRC interno del radio. Es importante destacar que el proceso de encriptado fue desactivado inicialmente para evitar problemas durante la configuración.

También se decidió activar el sistema interno de eliminación de factor DC del radio mediante el uso del método de Whitening o Scrambling[8]: el paquete a enviar y su CRC pasan por un proceso de aleatorización que lo transforman en una cadena de caracteres pseudo-aleatorios, lo que permite eliminar cadenas de bits del mismo tipo, disminuyendo el factor DC que pueda contener. En el receptor el mensaje pasa por el proceso inverso recuperando el mensaje. Es un proceso transparente para el usuario.

La configuración de potencia de salida del radio fue escogida activando el amplificador de potencia PA1, con un valor de nivel de potencia por defecto de 31, lo que ofrece una potencia de salida de 13dBm. El nivel de potencia puede ser modificado por software posteriormente.

### 5.3.2 Función RF\_ReConfig()

Si en algún momento se desea modificar la velocidad de transmisión del radio, el tamaño del paquete o el nivel de poder el radio se puede emplear la función RF\_ReConfig(). La función toma los valores de entrada y modifica el vector de configuración de manera pertinente. Una vez realizado el cambio reenvía el vector de configuración al radio y lo coloca en modo de “RX”, listo para recibir paquetes con la nueva configuración.

### 5.3.3 Función RF\_Mode\_ChangeTo()

La función RF\_Mode\_ChangeTo() permite cambiar de modo al radio. Toma como entrada el modo al que se desea cambiar, que puede ser: “Standby” para el modo de espera, “RX” para el modo de recepción, “TX” para el modo de transmisión, “Sleep” para el modo de bajo consumo y sintetizador de frecuencia. Inicialmente la función se asegura de que el radio no se encuentra cambiando de modo, una vez asegurada de esto, realiza el cambio de modo mediante la modificación del registro RegOpMode. Para el caso de los modos de Standby, Sleep y el sintetizador de frecuencia solo es necesaria la modificación del RegOpMode, pero para el caso del TX y RX se realizan además las configuraciones de los amplificadores del radio y el mapeo del pin DIO0.

El RFM69HW cuenta con dos amplificadores: PA1 y PA2. Durante la recepción los dos amplificadores deben ser desactivados y se debe colocar el registro RegTestPaX en modo de baja potencia, mientras que durante la transmisión se debe activar al menos uno de los amplificadores y se debe colocar el registro RegTestPaX en modo de alta potencia.

Con respecto al mapeo del pin DIO0, durante la transmisión se programa para que se active con la interrupción de paquete enviado, que ocurre en el momento en el que se envía el último byte del CRC, es decir, cuando se termina de enviar el paquete. Durante la recepción se programa para activarse con la interrupción de paquete recibido, que ocurre en el momento en el que el paquete se encuentra disponible en la memoria del radio, con acceso de tipo FIFO, después de haber pasado por la verificación del CRC interno del radio y haber descriptado el paquete.

#### 5.3.4 Función RF\_Send()

Al llamar a esta función, utilizando como entrada la dirección del vector donde está almacenado el paquete que se desea enviar, el radio pasa al modo de Standby, se envían todos los elementos del paquete a la FIFO del radio y el radio se pasa al modo de transmisión TX.

Debido a la configuración establecida, la transmisión del paquete inicia en el momento en el que el radio pasa a TX, por lo que el radio permanece en este modo hasta que el microcontrolador le indique lo contrario. El paquete a enviar, y por lo

tanto el vector, deben ser de exactamente el tamaño establecido durante la configuración del radio, de otro modo pueden ocurrir errores durante la transmisión: si se intenta enviar un paquete más pequeño el radio llenara la FIFO de información no deseada, perdiéndose el paquete, y si se intenta enviar un paquete más grande, introduciendo en la FIFO más elementos de los permitidos entonces al pasar a TX el radio funcionará de manera incorrecta, no enviará el mensaje y por consiguiente no se activará la interrupción de paquete enviado.

#### 5.3.5 Función RF\_PacketRecived()

En el momento en el que se reciba un paquete se puede llamar a la función RF\_PacketRecived() para extraerlo de la FIFO del radio y almacenarlo en el vector RF\_Packet[] para su posterior manejo. Esta función pasa el radio al modo Standby, extrae todos los elementos de la FIFO hasta que se encuentre vacía y regresa el radio al modo de RX. Para la configuración establecida tanto el vector RF\_Packet[] como el número de elementos extraídos siempre será igual al tamaño del paquete fijado.

#### 5.3.6 Función RF\_ReadRSSI()

La función RF\_ReadRSSI() permite leer el valor del registro RegRssiValue medida por el radio y almacenarla en la variable global RSSI. El proceso de medición de la RSSI del radio ocurre de manera automática en el momento en el que este detecta la señal de preamble de un mensaje, por lo cual la medición de la RSSI debe efectuarse lo más pronto posible después de recibir el paquete. Se recomienda usarla en el instante en el que ocurra la interrupción de paquete recibido.

La RSSI del radio tiene una resolución de 0.5 dBm, y se almacena en el registro RegRssiValue. Para obtener el valor de la RSSI a partir del RssiValue se emplea la Fórmula 3 [6]:

$$RSSI = \frac{-RssiValue}{2} \quad (3)$$

La variable global RSSI almacena directamente el valor del RssiValue.

#### 5.3.7 Función RF\_Encrypt()

RF\_Encrypt() se encarga de activar o desactivar el proceso de encriptado del radio. Si se llama a la función sin ninguna entrada entonces se desactiva el encriptado, pero si se llama con la dirección de la clave de encriptado entonces la clave se envía a los registros correspondientes y se activa el encriptado. La clave debe ser estrictamente de 16 bytes, en caso contrario se desactiva el proceso de encriptado para evitar errores.

#### 5.3.8 Función RF\_Debug()

Esta función lee los 84 registros del radio, los guarda en el vector RF\_DebugVec y los envía por el UART2 del microcontrolador para que el programador pueda analizarlos y determinar posibles fallas en la programación del radio.

### 5.3.9 Definiciones y valores iniciales

Debido a que el manejo del radio se lleva a cabo mediante la lectura y escritura de registros en tramas de comunicación de 16 bits, se diseñó un archivo donde se definen todas las direcciones de los registros y sus posibles valores con los nombres asociados a sus respectivas funciones de acuerdo a la hoja de datos del dispositivo. De esta manera, por ejemplo, la dirección 0x00 asociada a la FIFO del radio queda definida como RegFifo, y cada vez que se quiera acceder a esa dirección se empleara ese nombre. Esto facilita no solo la programación del código, sino su posterior lectura y modificación.

### 5.4 Estado 0: Espera

Una vez inicializados el microcontrolador y el radio la máquina de estados inicia su operación en el Estado 0. En este estado la máquina verifica la llegada de un carácter por el RS485 o de un paquete por el radio. En la Figura 10 se presenta un diagrama de flujo simplificado de la rutina del estado.

Al ocurrir la interrupción de Rx del UART2 del microcontrolador se activa la FlagRx, que indica la llegada de un carácter por el RS485, cuando la FlagRx es activada entonces la máquina pasa al Estado 1. En el caso de recibir un paquete por el radio entonces es la interrupción externa 1 del microcontrolador la que da la alerta, activando la bandera RF\_DIO0, informando a la máquina para que cambie al Estado 3.

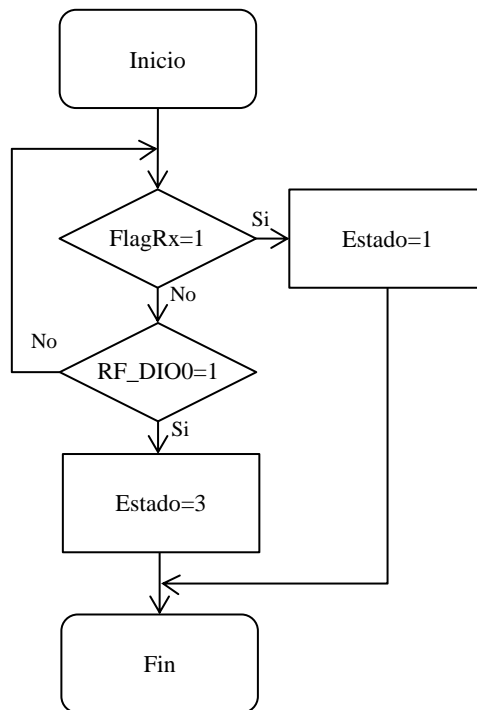


Figura 10: Diagrama de flujo simplificado del Estado 0.

### 5.5 Estado 1: Recepción por el RS485

El Estado 1 es el encargado de recibir la trama MODBUS por el RS485. El tamaño máximo de la trama que puede ser recibida está limitado a 1000 bytes. Cada vez que ocurre una interrupción de Rx del Uart2, es decir, cada vez que llega un carácter por el RS485, la rutina se asegura que quede espacio disponible y guarda el carácter en el buffer de datos DataBuff[]. En caso de no haber espacio en el buffer ocurre un error, descartando el carácter recibido.

Al entrar por primera vez en el Estado 1 el radio se pasa al modo de Standby para evitar la llegada de paquetes mientras se procesa la trama MODBUS. Se tomó esta consideración bajo la premisa de que en una red MODBUS RTU solo puede

haber un y solo un mensaje viajando por la red al mismo tiempo, de tal manera de que el dispositivo de puerta de enlace le da prioridad al primer mensaje que reciba, ya sea por el RS485 o por el radio, y cualquier otro mensaje es descartado automáticamente.

El Timer1 del microcontrolador es iniciado al entrar en el Estado 1 con un periodo de 3ms para actuar como el periodo de silencio necesario para recibir la trama. Cada vez que ocurre una interrupción de carácter recibido el contador del timer se reinicia, de tal manera que la interrupción del Timer1 ocurre únicamente cuando pasan 3ms sin que se reciba un carácter por el RS485.

En el momento en que ocurre la interrupción del Timer1 sin que ocurriera un error se asume que se recibió la trama completa, y se envía la máquina al Estado 2. En caso de haber ocurrido un error la trama completa se descarta, el radio se pasa a modo de RX y se envía la máquina al Estado 0.

En la Figura 11 se muestra el diagrama de flujo simplificado del Estado 1.

Es importante notar que, si en algún momento se recibe un carácter por el RS485 y la máquina no se encuentra en el Estado 0 o en el Estado 1 entonces todos los caracteres recibidos son descartados hasta que ocurra una nueva interrupción del Timer1. En caso de que se requiera enviar un mensaje por el RS485 mientras esto ocurre la máquina espera la interrupción antes de proceder a enviar la trama.



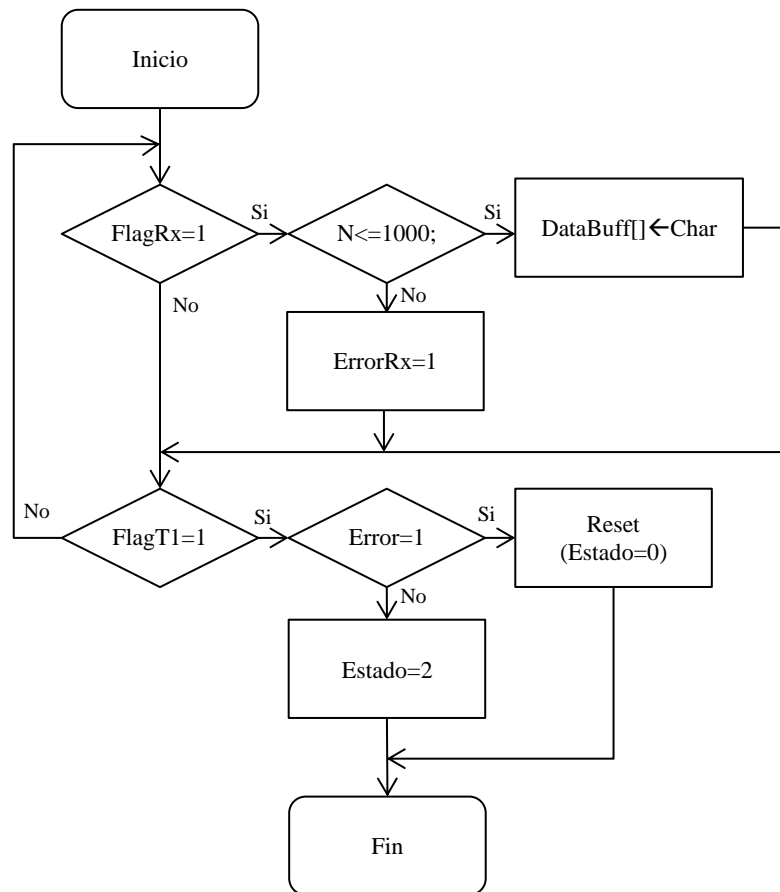


Figura 11: Diagrama de flujo simplificado del Estado 1.

## 5.6 Estado 2: Procesado de la trama

Una vez que se recibe la trama completa por la interfaz RS485 o se reciben todos los paquetes por el radio entonces se ingresa al Estado 2. En este estado la máquina lee el buffer de datos y decide qué hacer con la trama MODBUS: enviarla por el RS485, enviarla por el radio o descartarla. En la Figura 12 se muestra el diagrama de flujo simplificado del Estado 2.

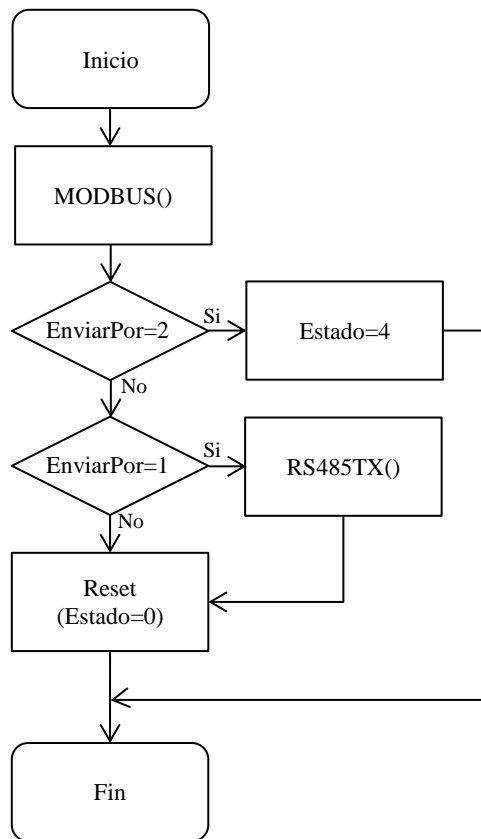


Figura 12: Diagrama de flujo simplificado del Estado 2.

Al ingresar en el Estado 2 se llama a la función MODBUS(), encargada de validar la trama y de cumplir con otras funciones a detallar en secciones posteriores de este tomo. La validación de la trama se realiza mediante la comprobación del CRC de la trama: Se calcula el CRC y se compara con el recibido, si es igual entonces la trama puede seguir su camino, en caso contrario la trama se descarta haciendo la variable EnviarPor=0, pasando la máquina al Estado 0 y pasando al radio a RX.

Si la trama fue recibida por el RS485 entonces la variable EnviarPor toma el valor 2 y la máquina pasa a Estado 4 para ser enviada por el radio. En caso de que la trama fuera recibida por el radio entonces EnviarPor toma el valor de 1, lo que llama

a la función RS485TX() que envía de forma continua el buffer de datos por el UART2 del microcontrolador y pasa a la máquina al Estado 0 para esperar por una nueva trama.

La función MODBUS() cumple con otra función particular para la realización de las pruebas de validación: fue programada para emplear el protocolo MODBUS para leer de forma remota la RSSI de los radios. Esta función solo fue implementada para las pruebas de validación, ya que emplea las ID de la 250 a la 252, reservadas por el protocolo. En el capítulo 7 se explica detalladamente las funciones agregadas para realizar las pruebas.

### 5.7 Estado 3: Recepción por el radio

Si se recibe un paquete por el radio mientras la máquina se encuentra en el Estado 0, esta pasará al Estado 3. Ya que las tramas MODBUS pueden ser mucho más grandes que los paquetes que maneja el radio se decidió separar las tramas para enviarlas parte por parte por el radio. Para esto se encapsularon en un protocolo propio mostrado en la Figura 13.

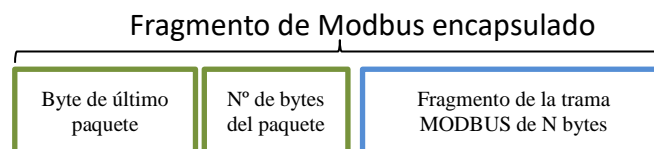


Figura 13: MODBUS encapsulado para el envío por el radio.

El primer byte del paquete toma el valor de 1 si es el último paquete que se está recibiendo o se va a enviar, y es 0 para otro caso. El segundo byte corresponde al número de caracteres de la trama MODBUS que se están enviando en ese paquete. De esa manera la máquina puede saber si debe esperar por más paquetes, si ya llegó el último, si debe tomar todos los bytes recibidos como parte de la trama o solo un fragmento.

Al entrar en el Estado 3 la máquina extrae el paquete de la FIFO del radio utilizando la función PacketRecived(), lo guarda en el buffer de datos y determina si es el último paquete. En caso de ser el último paquete la máquina pasa al Estado 2, en caso contrario la máquina espera hasta que llegue el próximo paquete en el Estado 3, iniciando el proceso nuevamente. En la Figura 14 se muestra el diagrama de flujo simplificado para el Estado 3.

Debido a que el envío de información por el radio no es infalible, y para evitar que la máquina quede atrapada en un ciclo esperando un último paquete que podría no llegar, se decidió implementar un tiempo de espera máximo entre paquetes mediante el uso de la interrupción del Timer2 del microcontrolador. Al momento de esperar por un paquete se inicia el Timer2, y si ocurre la interrupción antes de recibir un nuevo paquete la máquina asume que ocurrió la pérdida del paquete, descartando toda la trama recibida y pasando al Estado 0.

El tiempo en el que demora un paquete en ser recibido depende tanto de la velocidad de transmisión como del tamaño del paquete, así como del tiempo que demora el radio en manejar los paquetes y el microcontrolador en procesar la trama. Tomando en cuenta que para la velocidad más baja de 1200bps, y para el tamaño de paquete más grande de 64 bytes el tiempo en el aire del paquete es de

aproximadamente 0.5 segundo, se decidió tomar 750ms como periodo límite para el Timer2.

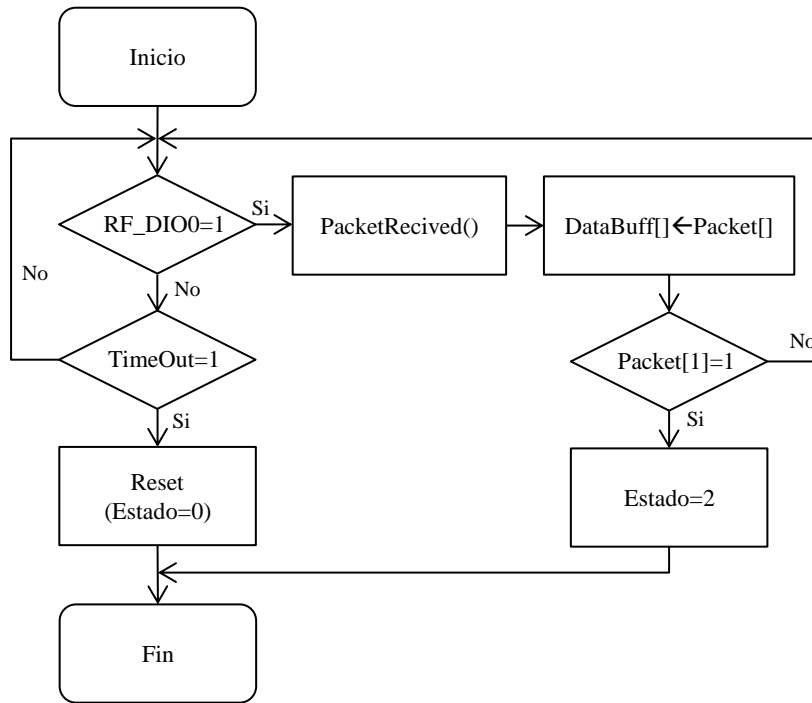


Figura 14: Diagrama de flujo simplificado del Estado 3.

### 5.8 Estado 4: Envío por el radio

En el Estado 4 la trama almacenada en el buffer de datos es enviada por el radio. En la Figura 15 se muestra el diagrama de flujo simplificado de este estado. Como se mencionó en la sección 5.7 para poder enviar la trama por el radio fue necesario encapsular fragmentos de la trama MODBUS en un protocolo propio, descrito previamente en la Figura 13.

Al ingresar en el Estado 4 se toma la trama del buffer de datos y se guarda en el vector de envío Packet[] se llama la función PacketSend() y esta procederá a guardar el paquete en la FIFO del radio y lo enviará. Para evitar posibles errores de comunicación se implementó un tiempo de espera máximo para el envío de paquete mediante el uso del Timer2, del mismo modo que se hizo para la recepción de paquetes. Si ocurre esta interrupción del Timer2 toda la trama es descartada, el radio es reprogramado desde su configuración inicial y la máquina regresa al Estado 0.

Si el tamaño de la trama MODBUS es mayor al tamaño del paquete el primer byte del vector de envío se hace cero, lo que indica que faltan paquetes por enviar. En este caso el sistema debe esperar un tiempo antes de enviar el siguiente paquete. Ya que el tiempo en el aire es el mismo tanto para el receptor como para el transmisor, entonces el tiempo de espera solo debe contemplar el tiempo en que el radio receptor procesa el paquete y el microcontrolador del radio receptor lo saca de la FIFO del radio.

Una vez enviados todos los paquetes la máquina regresa a Estado 0 para esperar una nueva trama.

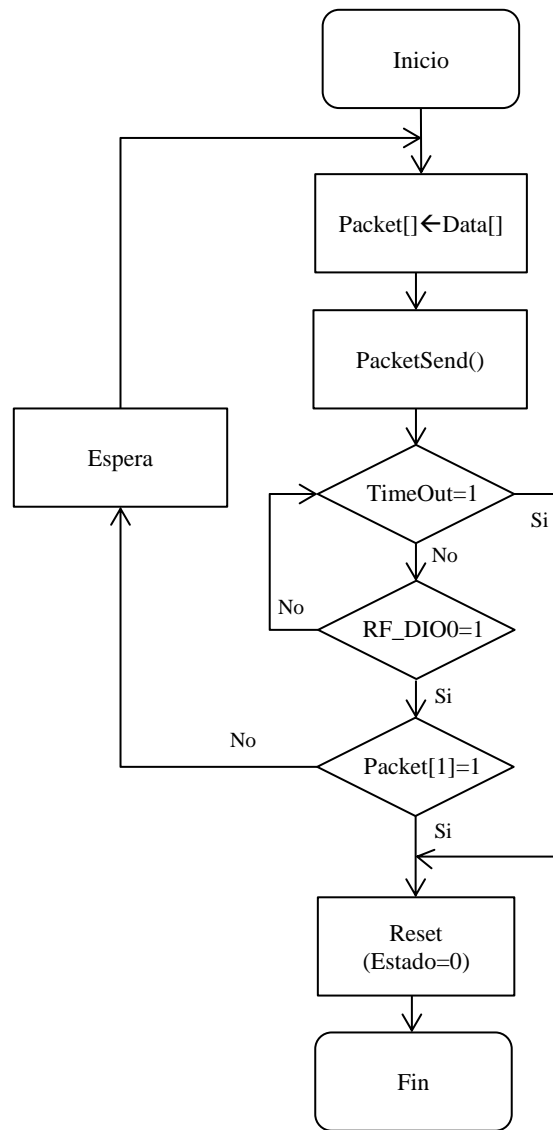


Figura 15: Diagrama de flujo simplificado del Estado 4.

## CAPÍTULO VI

### VALIDACIÓN DE LOS DISPOSITIVOS

#### 6.1 Consideraciones previas para la validación

Como se mencionó en el Capítulo 5 se realizaron modificaciones en la función MODBUS() para facilitar la realización de las pruebas de campo. Las modificaciones realizadas permiten utilizar el programa Modbus Poll v5.0.1 de Witte Software para interrogar a los dispositivos como si fueran esclavos MODBUS. El Modbus Poll es un simulador de maestros MODBUS con soporte para MODBUS TCP/IP y MODBUS RTU/ASCII.

En la Tabla 8 se describen las funciones MODBUS implementadas para la realización de las pruebas de validación. Cualquier petición que no se encuentre entre las funciones implementadas regresa la excepción correspondiente.

Para facilitar la visualización de la información en el Modbus Poll las direcciones de los registros no siguen un orden lineal. A continuación se procede a explicar el funcionamiento de las funciones implementadas:

- Función 3 con direcciones de la 0 a la 3: permite leer el ID, BitRate, PowerLevel y PacketLength del dispositivo.



- Función 3 con direcciones de la 150 a la 152: realiza la medición de la RSSI del dispositivo solicitado. Cuando la trama llega al dispositivo seleccionado este regresa el número de paquetes enviados y su ID, la trama regresa al dispositivo conectado al maestro y es este el que determina la RSSI del mensaje recibido y la coloca en la trama (realizando todas las modificaciones pertinentes como el nuevo cálculo del CRC). De esta manera la medición de la RSSI se realiza de los esclavos al maestro, y el número de paquetes perdidos puede ser determinado al comparar los mensajes recibidos con los mensajes enviados por el dispositivo solicitado.

Función	Dirección	Descripción
3	0-3	Lee el ID, BitRate, PowerLevel y PacketLength actuales.
	150-152	Lee la última RSSI medida, el número de paquete enviados y el ID del dispositivo.
	100-200	Lee las últimas 100 RSSI medidas.
16	0-4	Escribe el Edit on/off, los nuevos valores de BitRate, PowerLevel y PacketLength, y el RSSI clear.

Tabla 8 : Funciones MODBUS implementadas para la validación.

- Función 3 con direcciones de la 100 a la 200: permite leer las últimas 100 mediciones de la RSSI, facilitando la adquisición de la información. Si se realizan más de 100 mediciones el dispositivo descarta la más antigua guardando las más recientes. Los valores de RSSI son guardados en el RssiVector[].
- Función 16 con direcciones de la 0 a la 4: Escribe los nuevos valores del Edit on/off, BitRate, PowerLevel y PacketLength, y el RSSI clear. Los dispositivos

aceptan todos los valores de BitRate de la Tabla 7, el PowerLevel puede variar entre 0 y 31 y el PacketLength debe estar comprendido entre 3 bytes y 64 bytes, cualquier otro valor es ignorado por los dispositivos. Escribir 1 en el RSSI clear limpia los valores del RssiVector[]. El Edit on/off permite habilitar o deshabilitar la modificación del resto de los registros, esto para evitar modificar erróneamente alguno de ellos durante las pruebas. Si el Edit on/off es 0 entonces no se realizará ningún cambio en los registros, en el momento en el que cambie a 1 todas las modificaciones serán realizadas.

En la Figura 16 se presenta una captura del programa de validación en ejecución, donde se muestra la conexión con 3 dispositivos: el Slave0 que corresponde a una de las puertas de enlace prototipo (PEv1) conectada por RS485 al maestro MODBUS, el Slave1 como un segundo prototipo conectado mediante el radio a la red y el Slave 2 como la versión final del dispositivo (PEv2) conectado a la red mediante el radio. Ya que a nivel de transmisión por radio la única diferencia entre las versiones de los dispositivos es el tipo de antena, esta configuración ofrece una manera de comparar el desempeño de los distintos tipos de antenas usadas. La misma configuración fue utilizada para todas las pruebas de validación. En la sección de ANEXOS se presentan las tablas con los datos empleados para construir las figuras de este capítulo.

Para el caso de la Figura 16, se observa el monitoreo de las tres puertas de enlace en las ventanas SlaveStatus0-2, el vector de últimas 100 RSSI medidas en la ventana RssiVector, las ventanas de edición de parámetros de las puertas de enlace denominadas SlaveEdit0-2 y la ventana de ReadRSSI que presenta la información de la última RSSI medida, junto con el número de paquetes enviados por la puerta de enlace que realizó el envío del paquete y el ID de este dispositivo.

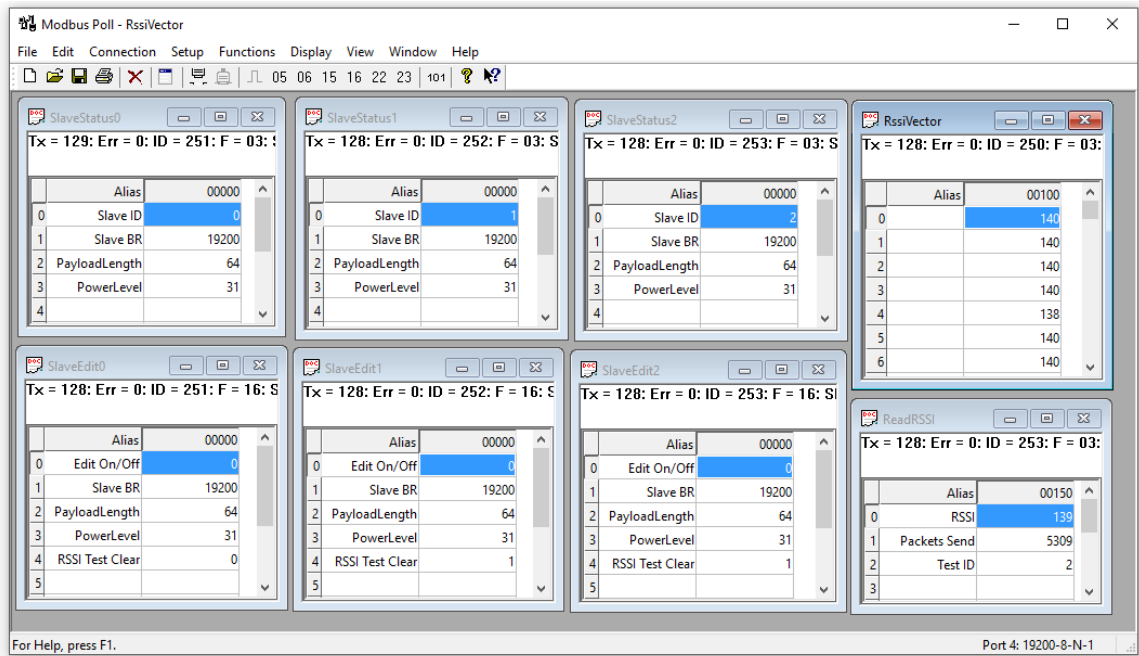


Figura 16: Captura de pantalla del programa de validación.

## 6.2 Prueba 1: Análisis de propagación con línea de vista

La primera prueba realizada fue el análisis de propagación con línea de vista, con el objeto de determinar el comportamiento de los dispositivos y comprobar la pérdida de paquetes al incrementar la distancia. El comportamiento esperado para esta prueba es una disminución de la RSSI al incrementar la distancia de las mediciones.

La prueba se realizó en el edificio residencial Los Carrizales en San Antonio, que cuenta con un balcón en el piso 13 con línea de vista a varios puntos adecuados para realizar las mediciones. Se emplearon dos dispositivos: un prototipo PEv1 y un dispositivo PEv2. A fines prácticos estos dispositivos actúan como esclavos

MODBUS a ser interrogados por el maestro. En el lado del maestro se colocó un prototipo PEv1. Los dos esclavos se colocaron en el balcón del edificio y el maestro se movió por los distintos puntos de tal manera de realizar el análisis de propagación para las dos versiones diseñadas.

Las mediciones se realizaron a distancias aproximadas de 1m (como referencia), 20m, 50m, 100m, 150m y 385m. Se hicieron peticiones para medir la RSSI hasta obtener 100 mediciones en cada punto para las cuatro velocidades programadas. La RSSI se determinó promediando las 100 mediciones obtenidas. En la Figura 17 se presentan los resultados obtenidos para las mediciones de la RSSI, y en la Tabla 9 el porcentaje de paquetes perdidos obtenido para cada caso.

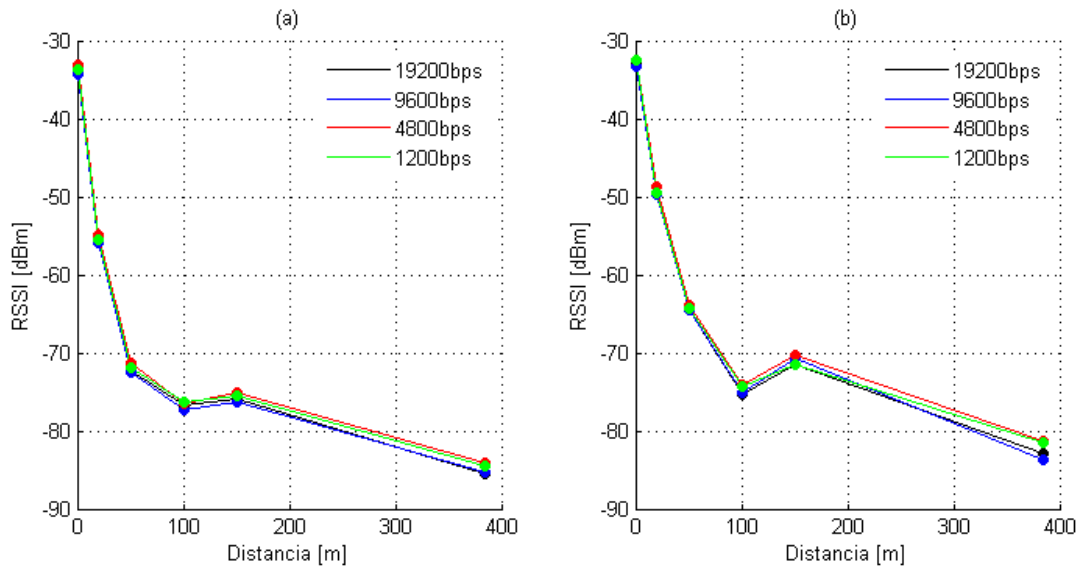


Figura 17: (a) RSSI vs Distancia para la PEv1. (b) RSSI vs Distancia para la PEv2.

Como se puede observar en la Figura 17 el valor de la RSSI al incrementar la distancia fue disminuyendo, como era esperado. Adicionalmente se pudo comprobar

que la comunicación no se interrumpió para ninguno de los casos. La diferencia en el desempeño para las distintas velocidades fue muy similar para todos los casos.

Es importante notar como se observa un patrón en el que a mayores distancias, las velocidades de 4800bps y 1200bps se desempeñan hasta 2dBm mejor que para las velocidades de 19200bps y 9600bps, en particular para el PEv2. Lo que podría indicar un alcance superior para estas velocidades.

	Velocidad [bps]	Distancia [m]					
		1	20	50	100	150	385
PEv1	19200	0,0	0,0	0,0	0,0	0,0	0,0
	9600	0,0	0,0	1,0	0,0	0,0	0,0
	4800	0,0	0,0	0,0	1,0	0,0	0,0
	1200	0,0	0,0	0,0	0,0	2,0	0,0
PEv2	19200	0,0	0,0	0,0	0,0	0,0	0,0
	9600	0,0	0,0	0,0	0,0	0,0	0,0
	4800	0,0	0,0	0,0	0,0	0,0	0,0
	1200	0,0	0,0	0,0	0,0	1,0	0,0

Tabla 9: Porcentaje de paquetes perdidos [%] para la Prueba 1.

La comparación entre velocidades para ambos dispositivos se presenta de manera más clara en la Figura 18, donde se observa un mejor desempeño en todos los casos para el PEv2.

El porcentaje de error obtenido para la Prueba 1 se encuentra dentro de lo aceptable, en particular al observar la Tabla 10 donde se presenta el porcentaje de error promedio para cada velocidad, en el que en ningún caso el error supera el 0.3% de paquetes perdidos.

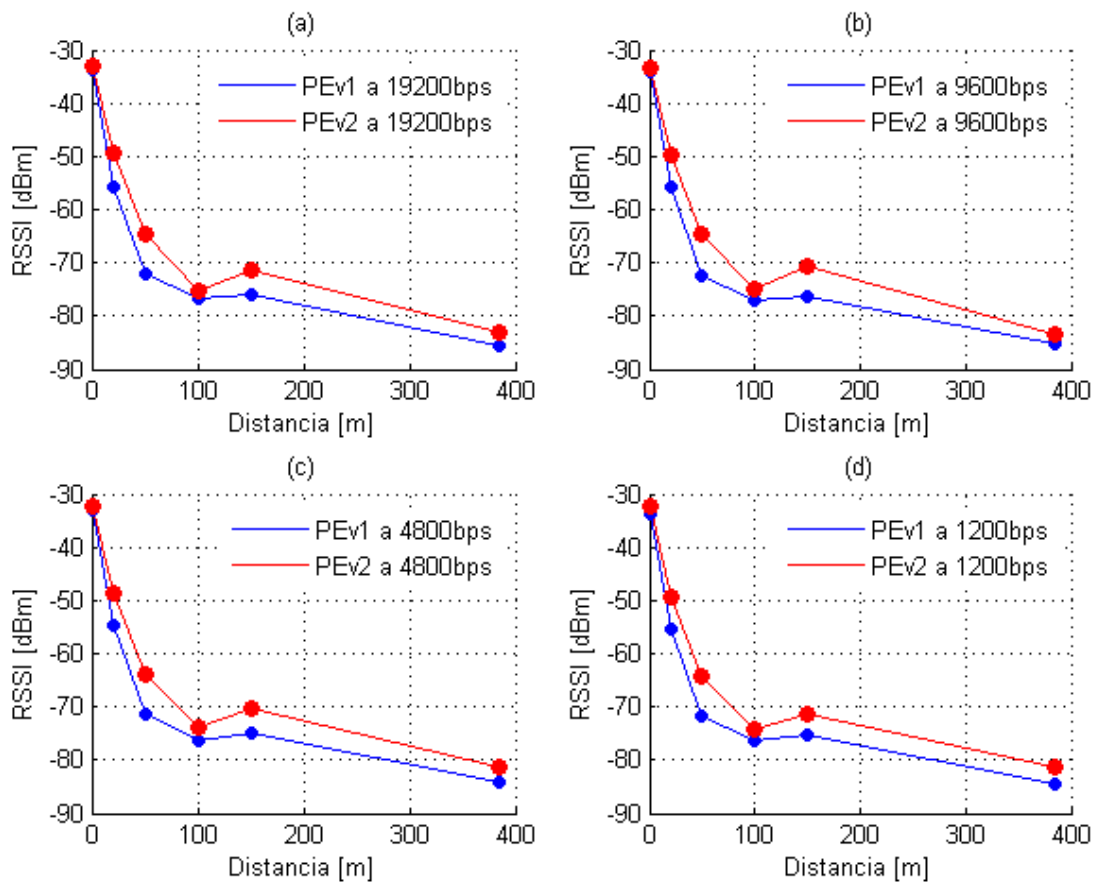


Figura 18: (a) RSSI vs Distancia a 19200bps. (b) RSSI vs Distancia a 9600bps. (c) RSSI vs Distancia a 4800bps. (d) RSSI vs Distancia a 1200bps.

	Velocidad [bps]	Porcentaje de error promedio [%]
PEv1	19200	0,0
	9600	0,2
	4800	0,2
	1200	0,3
PEv2	19200	0,0
	9600	0,0
	4800	0,0
	1200	0,2

Tabla 10: Error promedio para cada velocidad para la Prueba 1.

### **6.3 Prueba 2: Análisis de propagación en un ambiente controlado**

El objeto de esta prueba fue realizar un análisis de propagación en un ambiente controlado para determinar el comportamiento de los dispositivos diseñados dentro de una edificación. El ambiente consistió en un edificio residencial de 8 pisos.

Para realizar la prueba se colocaron dos dispositivos en el piso 8, una PEv1 y una PEv2, como esclavos MODBUS y se procedió a medir la RSSI de los dispositivos descendiendo piso a piso con el maestro MODBUS conectado a un dispositivo PEv1 hasta que se detuvo la comunicación. Se enviaron peticiones hasta conseguir 100 mediciones de RSSI para cada piso y se realizaron pruebas para las 4 velocidades programadas. La RSSI se determinó promediando las 100 mediciones obtenidas.

En la Figura 19 se presentan los resultados obtenidos para la medición de la RSSI, y en la Figura 20 se puede observar el porcentaje de paquetes perdidos para cada caso estudiado. La comunicación no se perdió para ninguna de las pruebas. Para poder facilitar la presentación de la información y la comprensión de los resultados, se asumió el piso 8 como piso 0, y al descender cada piso se le colocó una numeración ascendente como etiqueta en las figuras de resultados.

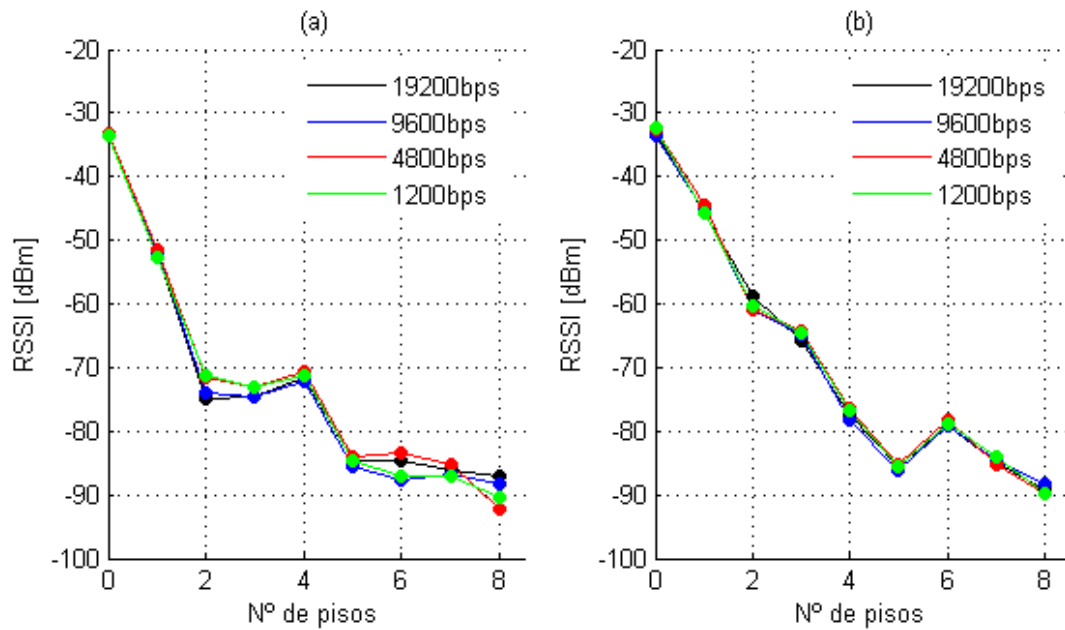


Figura 19: (a) RSSI vs N° de pisos para la PEV1. (b) RSSI vs N° de pisos para la PEV2.

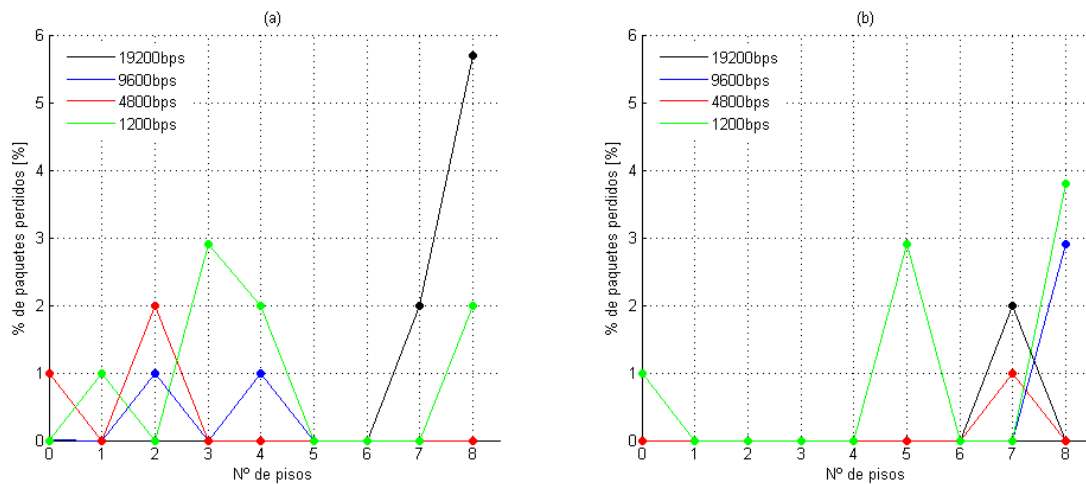


Figura 20: (a) Porcentaje de paquetes perdidos vs N° de pisos para la PEV1. (b) Porcentaje de paquetes perdidos vs N° de pisos para la PEV2.



El porcentaje de pérdida de paquetes está dentro de lo aceptable, en particular al observar la Tabla 11, donde se presenta el % de paquetes perdidos promedio para cada velocidad. Para ninguno de los casos el error supera el 1%, aunque se puede ver que a los 1200bps se presenta la mayor pérdida de paquetes. Si bien para la PEv2 se puede hacer la relación de mayor pérdida de paquetes a mayor distancia, esta relación no se cumple para la PEv1, en la que se presenta una pérdida de paquetes más constante para todos los pisos.

	Velocidad [bps]	Porcentaje de Error promedio [%]
PEv1	19200	0,8
	9600	0,4
	4800	0,3
	1200	0,9
PEv2	19200	0,2
	9600	0,3
	4800	0,1
	1200	0,9

Tabla 11: Porcentaje de error promedio para las distintas velocidades.

Al comparar la intensidad de señal para cada una de las velocidades no se pudo notar una diferencia clara, ya que el comportamiento para todos los casos fue muy similar. Esto indica que para los pisos en donde se realizaron las pruebas, la velocidad de transmisión no afecta la RSSI de los dispositivos.

Para notar de manera más clara las diferencias entre el comportamiento de las RSSI de las dos versiones de puertas de enlace, se presenta en la Figura 21 una comparación entre las mediciones realizadas para cada velocidad.

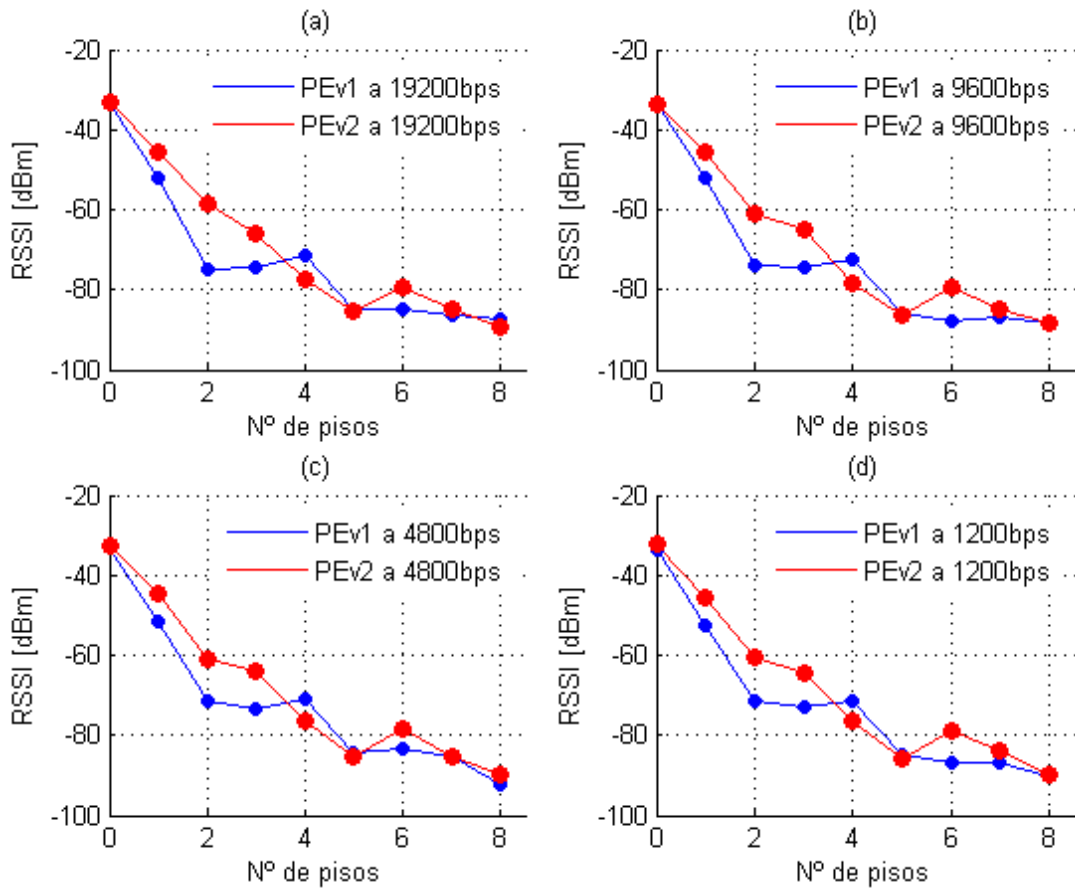


Figura 21: (a) RSSI vs N° de pisos a 19200bps. (b) RSSI vs N° de pisos a 9600bps. (c) RSSI vs N° de pisos a 4800bps. (d) RSSI vs N° de pisos a 1200bps.

Al analizar la Figura 21 se puede ver como la versión final del dispositivo (PEv2) tiende a presentar una mejor RSSI para la mayor parte de los pisos, pero la PEv1 parece igualar su desempeño al llegar al último piso para todas las velocidades. En ningún momento ocurrió la pérdida de comunicación entre los dispositivos.

### **6.4 Prueba 3: Implementación de la red en condiciones reales**

Una vez realizados los análisis de propagación se decidió realizar una prueba simulando condiciones de operación reales para determinar la confiabilidad de la red diseñada. Con esta prueba se espera determinar si la red es capaz de establecer comunicación continua en un ambiente real, para esto se pretende determinar el porcentaje de error promedio al final de la prueba y comprobar sé que encuentre en el rango de valores aceptados para una red inalámbrica. Para esto se decidió implementar una red MODBUS RTU con dos terminales remotas y un servidor en un ambiente sin línea de vista. El ambiente de pruebas consistió en un conjunto de oficinas ubicadas en el primer piso del Hospital San Juan de Dios en Caracas.

La primera terminal remota (RTU1) consiste en una terminal que se comunica a 9600 baudios, con ID 1 y que tiene la capacidad de realizar mediciones de temperatura y humedad. La segunda terminal remota (RTU2) se comunica a 19200 baudios, con ID 2 y con la capacidad de medir la temperatura. El servidor consiste en un computador portátil con Modbus Poll para interrogar a las remotas que se comunica con la puerta de enlace a 19200 baudios. Cada uno de los dispositivos fue conectado a una de las puertas de enlace diseñadas: uno de los prototipos conectado al maestro, otro de los prototipos conectado a la RTU2 y el dispositivo final conectado a la RTU1. En la Figura 22 se presenta un diagrama básico con la configuración de la prueba. El maestro se encuentra en una oficina, mientras que la RTU2 se encuentra en otra oficina a aproximadamente 20 metros, la RTU 1 se encuentra en otra oficina adyacente a esta.

En el aire la red se comunica a 19200 bps, el tamaño de los paquetes se definió en 64 bytes y el nivel de potencia se fijó en 13dBm. La prueba se llevó a cabo

interrogando a las remotas con la función 3 de MODBUS cada 5 segundos, durante un periodo de aproximadamente 2 días. Los resultados obtenidos se presentan en la Tabla 12.

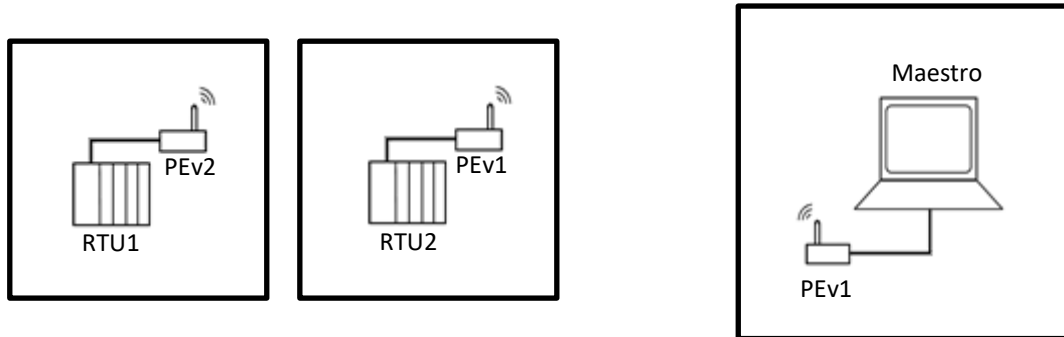


Figura 22: Configuración general para la prueba 3.

ID de la remota	Paquetes enviados	Paquetes perdidos	RSSI [dBm]	Porcentaje de paquetes perdidos [%]
1	37540	578	-75	1,54
2	37540	41	-65	0,11

Tabla 12 : Resultados de la implementación en condiciones reales.

Como se puede observar en la Tabla 12 el porcentaje de error para la puerta de enlace 1 fue del 0.11%, y de 1.54% para la puerta de enlace 2. En ambos casos la comunicación se mantuvo estable durante toda la duración de la prueba, y si bien en el segundo caso el error es un poco superior al obtenido en los análisis de propagación de las pruebas 1 y 2, sigue estando dentro de los valores aceptables para una red inalámbrica.

## CONCLUSIONES

Con la elaboración de este trabajo especial de grado se demostró la importancia que tienen las redes de comunicación inalámbricas. Se cumplió con los objetivos planteados y se recopiló información respecto a las distintas tecnologías de comunicación inalámbrica y al protocolo MODBUS RTU. Además, se pudo validar la selección del RFM69HC/HCW como dispositivo de comunicaciones inalámbricas en la banda de 433MHz, y se diseñó una librería en C para PIC, capaz de configurarlo y programarlo, que puede ser utilizada en otras aplicaciones.

Como se planteó en el Capítulo III la red implementada resulta completamente transparente para el usuario, por lo que desde el exterior la red MODBUS RTU cumplirá con la topología con la que fue diseñada inicialmente, normalmente una topología de bus, mientras que en el aire la red diseñada actúa con un bus de datos, conectando a todos los nodos inalámbricos y comunicándose entre ellos.

El diseño de las puertas de enlace sirvió para establecer la red inalámbrica MODBUS RTU industrial, ya sea creando una red inalámbrica nueva, agregando nuevos elementos o hasta interconectando de manera inalámbrica nuevas secciones con la red preexistente.

Las puertas de enlace diseñadas tienen la capacidad de comunicarse con cualquier dispositivo MODBUS RTU, con una velocidad de 19600Bd o 9600Bd, además tienen la capacidad de trabajar con cuatro velocidades de transmisión en el aire, 19200bps, 9600bps, 4800bps y 1200bps, todas independientes de la velocidad de transmisión de la red MODBUS original.

Las pruebas de validación realizadas permitieron determinar el análisis de la propagación de los dispositivos, estableciendo el alcance de la red en 385m con línea de vista para las cuatro velocidades, además de corroborar su funcionamiento al ser implementada sin línea de vista y en edificaciones.

Ya que las pruebas no se pudieron realizar hasta que ocurriera la pérdida de comunicación, no se puede concluir cuál de las dos versiones de hardware presenta mejor alcance, pero se pudo notar un desempeño superior para la versión final del dispositivo en comparación a los prototipos iniciales para las Pruebas 1 y 2.

Con la Prueba 2 se validaron los dispositivos diseñados en el ambiente controlado, presentando un alcance mínimo de 8 pisos en condiciones similares al ambiente controlado de la prueba, con una pérdida de paquetes promedio menor al 1% para todas las velocidades probadas.

Los porcentajes de pérdida de paquetes para todas las pruebas resultan satisfactorios, validando los dispositivos y demostrando que son capaces de establecer una red MODBUS RTU inalámbrica estable y funcional en condiciones reales.

## RECOMENDACIONES

Los módulos RFM69HC/HCW cumplieron satisfactoriamente con los objetivos planteados, pero no son los únicos dispositivos de RF del mercado, en la actualidad existen gran variedad de dispositivos, incluyendo microcontroladores con dispositivos RF integrados como el ARM Cortex-M3. Se sugiera la evaluación de estos dispositivos para la realización de futuras versiones del hardware que permitan simplificar el diseño de las puertas de enlace, disminuir su tamaño y mejorar el desempeño de la red.

Se sugiere considerar una topología de red tipo mesh capaz de interconectar los dispositivos. De esta manera no solo se podría mejorar el desempeño de la red, sino también incrementar la distancia que es capaz de cubrir al utilizar nodos de la red como repetidores para llegar a nodos muy alejados del maestro.

Tomando en cuenta los resultados obtenidos en relación al alcance y el desempeño en las pruebas de validación, se sugiere evaluar los dispositivos para distancias mayores con el objeto de determinar el alcance máximo con línea de vista, además se propone realizar un análisis de propagación enfocado en evaluar el desempeño con distintos tipos de antenas, para conseguir mejores distancias e incrementar la confiabilidad de los dispositivos.

## REFERENCIAS BIBLIOGRÁFICAS

[1] Stallings, W. Data and Computer Communications (8th ed.). New Jersey: Harlow: Pearson.

[2] Zimmermann, Hubert. OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection. IEEE Transactions on Communications, Vol 28 N° 4 p. 425 - 432 [Consulta 2017].

[3] Internetworking Basics. <<https://www.cisco.com/cpress/cc/td/cpress/fund/ith/ith-01gb.htm#xtocid166844>> [Consulta 2017].

[4] MODBUS Organization, MODBUS Application Protocol Specification V1.1b3 <[http://www.modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b3.pdf](http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf)> [Consulta 2017].

[5] National Instruments, Información Detallada sobre el Protocolo Modbus <<http://www.ni.com/white-paper/52134/es/>> [Consulta 2017].

[6] MODBUS Organization, MODBUS over Serial Line Specification and Implementation Guide V1.02 <[http://www.modbus.org/docs/Modbus\\_over\\_serial\\_line\\_V1\\_02.pdf](http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf)> [Consulta 2017].

[7] HOPERF Electronic. RFM69HCW ISM Transceiver Module V1.1. RFM69HCW. Hoja de datos. [Consulta 2017].

[8] Semtech. SX1231H Transceiver. SX1231H. Hoja de datos. [Consulta 2017].



## BIBLIOGRAFÍAS

Abinayaa, V., & Jayan, A. *Case Study on Comparison of Wireless Technologies in Industrial Applications*. International Journal Of Scientific And Research Publications, Vol 4 p.1-5, 2014.

HOPERF Electronic. *RFM69HC ISM Transreciver Module V1.3*. RFM69HC. Hoja de datos.

Kaur, N., & Monga, S. *Comparisons of a Wired and Wireless Network: A Review*. International Journal Of Advanced Engineering Technology, Vol 5 N°2, p.34-35, 2014.

Semtech. *SX1231 Transceiver*. SX1231. Hoja de datos.

## ANEXOS

[ANEXO N°1]

**Tablas con los datos utilizados para construir las gráficas del capítulo 7.**

A continuación se presentan las tablas con los valores de RSSI promedio medidos durante las pruebas de validación.

		Velocidad [bps]	Distancia [m]				
			1	20	50	100	150
PEv1	19200	-33,9	-55,8	-72,2	-76,7	-75,9	-85,4
	9600	-34,2	-55,8	-72,5	-77,2	-76,2	-85,3
	4800	-33,1	-54,8	-71,3	-76,4	-75,0	-84,1
	1200	-33,6	-55,4	-71,8	-76,2	-75,4	-84,4
PEv2	19200	-32,9	-49,4	-64,6	-75,2	-71,5	-82,9
	9600	-33,2	-49,6	-64,6	-75,0	-70,7	-83,6
	4800	-32,4	-48,7	-63,8	-74,0	-70,2	-81,2
	1200	-32,4	-49,4	-64,3	-74,2	-71,4	-81,4

Tabla 13: Tabla de datos de la medición de la RSSI [dBm] promedio para la Prueba 1.

		Piso								
		0	1	2	3	4	5	6	7	8
RTU 1	19200	-33,4	-52,1	-74,9	-74,5	-71,6	-84,6	-84,6	-86,2	-87,2
	9600	-33,5	-51,9	-74,0	-74,6	-72,2	-85,6	-87,7	-86,9	-88,4
	4800	-33,3	-51,6	-71,6	-73,2	-70,7	-84,1	-83,5	-85,3	-92,3
	1200	-33,5	-52,7	-71,4	-73,0	-71,2	-84,7	-87,0	-87,0	-90,4
RTU 2	19200	-33,0	-45,6	-58,7	-65,8	-77,4	-85,5	-79,2	-85,0	89,1
	9600	-33,5	-45,4	-60,8	-64,9	-78,2	-86,2	-79,2	-84,6	-88,4
	4800	-32,5	-44,6	-60,9	-64,2	-76,3	-85,2	-78,4	-85,3	-89,8
	1200	-32,4	-45,7	-60,4	-64,5	-76,6	-85,7	-78,8	-83,9	-89,7

Tabla 14: Tabla de datos con la medición de la RSSI [dBm] promedio para la Prueba 2.

A continuación se presentan los valores obtenidos de porcentaje de paquetes perdidos tras realizar la Prueba 2.

Piso		Porcentaje de paquetes perdidos [%]								
		0	1	2	3	4	5	6	7	8
PEv1	19200	0,0	0,0	0,0	0,0	0,0	0,0	0,0	2,0	5,7
	9600	0,0	0,0	1,0	0,0	1,0	0,0	0,0	0,0	2,0
	4800	1,0	0,0	2,0	0,0	0,0	0,0	0,0	0,0	0,0
	1200	0,0	1,0	0,0	2,9	2,0	0,0	0,0	0,0	2,0
PeV2	19200	0,0	0,0	0,0	0,0	0,0	0,0	0,0	2,0	0,0
	9600	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	2,9
	4800	0,0	0,0	0,0	0,0	0,0	0,0	0,0	1,0	0,0
	1200	1,0	0,0	0,0	0,0	0,0	2,9	0,0	0,0	3,8

Tabla 15: Tabla de datos con el porcentaje de paquetes perdidos para la Prueba 2.

[ANEXO N°2]

**Manual de operación.**

# MANUAL DE OPERACIÓN

para el dispositivo de puerta de enlace PEv2.XY para la implementación de una red MODBUS RTU inalámbrica

Realizado por Alejandro Temprano

El siguiente manual especifica el proceso de instalación del dispositivo de puerta de enlace PEv2.

## SECCIÓN 1: Descripción General del Dispositivo

El PEv2. se encarga de la comunicación inalámbrica entre distintos dispositivos que utilicen el protocolo de comunicación MODBUS RTU.

El dispositivo se comunica con el servidor y con los esclavos mediante un cable serial RS-485. Es alimentado por una fuente externa de alimentación de 5 voltios y la comunicación entre los distintos dispositivos se lleva a cabo en la banda de 434MHz.

El dispositivo incluye una antena con conector SMA macho.

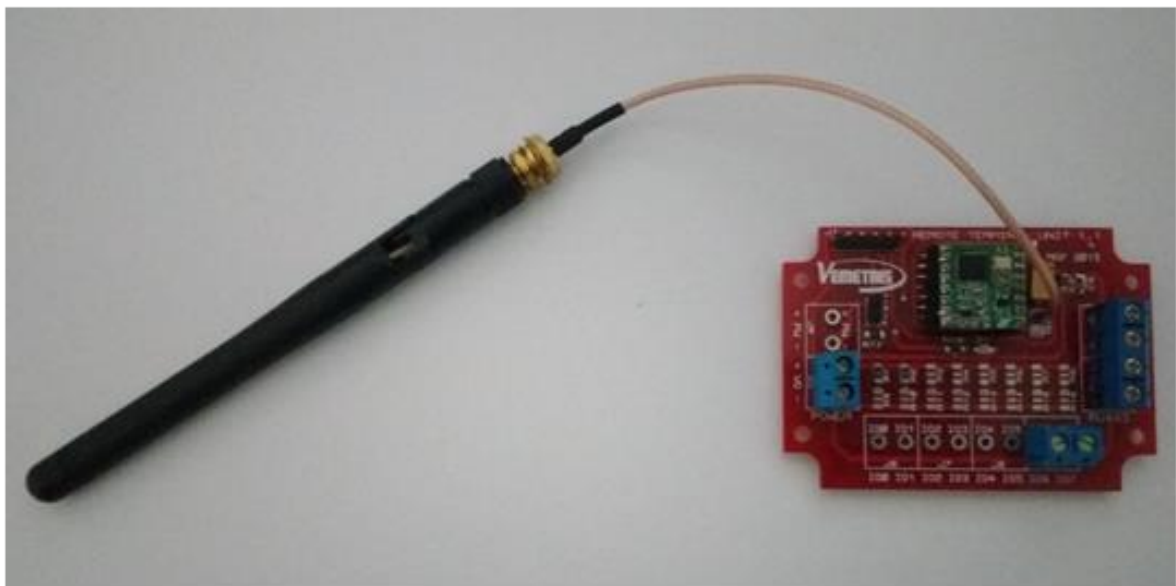


Figura 1: PEv2

---

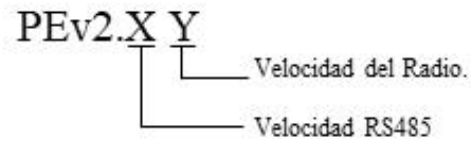
## SECCIÓN 2: Características del Dispositivo

---

Características básicas del dispositivo:

- Voltaje de alimentación: 5v.
- Interfaz RS-485.
- Velocidad de comunicación serial: 19200bps o 9600bps, dependiendo de la versión (Ver Tabla 1).
- Velocidad de transmisión en el aire: 19200bps, 9600bps, 4800bps o 1200bps, dependiendo de la versión (ver Tabla 2).
- Potencia de transmisión: +13dBm.
- Motor de paquetes incluido con encriptado AES-128bits, CRC y DC-Free.
- Rango con línea de vista: 385m.

Versión del dispositivo:



Versión X	Velocidad del RS485 [Bps]
1	19200
2	9600

Tabla 1: Velocidades disponibles del RS485.

Versión Y	Velocidad del Radio [bps]
1	19200
2	9600
3	4800
4	1200

Tabla 2: Velocidades disponibles del Radio.

---

## SECCIÓN 3: Conexión con el Dispositivo

---

La conexión con el dispositivo se lleva a cabo mediante un cable neutro, la alimentación y el par trenzado para la conexión con el RS-485 ( Ver Figura 2 ). La conexión es la siguiente:

- El cable neutro va al neutro (-) del PEv2, marcado en color negro.
- El A y el B del RS485 del dispositivo MODBUS al A (azul) y al B (verde) del PEv2.
- Se conecta la alimentación al positivo del dispositivo, marcado en rojo.
- Al alimentar el dispositivo el led se encenderá 2 veces por 1 segundo. Durante la transmisión o recepción el led se encenderá.



Figura 2: Conexión con el PEv2.