

Implementación de una red Modbus RTU inalámbrico en FPGA con aplicación en automatización de procesos

Autor: Rubén Dario Morales G.
Director de proyecto de Grado: PhD. José Luis Villa

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
MAESTRÍA EN INGENIERIA CON ENFASIS EN INGENIERÍA
ELECTRICA Y ELECTRÓNICA
CARTAGENA
Abril, 2011

Implementación de una red Modbus RTU inalámbrico en FPGA con aplicación en automatización de procesos

Rubén Darío Morales

Plan de Trabajo de Grado presentado como requisito para
optar al título de Magíster en Ingeniería.

Director de proyecto de Grado: PhD. José Luis Villa

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA
CARTAGENA
2011

Cartagena D. T. y C., 4 de Abril del 2011.

Señores

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

Facultad de Ingeniería Eléctrica y Electrónica

Comité Evaluador de Trabajos de Grado

Cordial saludo.

Por medio de la presente, me permito someter a su consideración el Trabajo de Grado titulado "Implementación de una red Modbus RTU inalámbrico en FPGA con aplicación en automatización de procesos" del estudiante RUBEN DARÍO MORALES GARCÍA, para optar por el grado de Magíster en Ingeniería con énfasis en Ingeniería Eléctrica y Electrónica.

Cordialmente,

RUBEN D. MORALES GARCIA

Cartagena D. T. y C., 4 de Abril del 2011.

Señores

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

Facultad de Ingeniería Eléctrica y Electrónica

Comité Evaluador de Trabajos de Grado

Cordial saludo.

Por medio de la presente, me permito someter a su consideración el Trabajo de Grado titulado "Implementación de una red Modbus RTU inalámbrico en FPGA con aplicación en automatización de procesos", del estudiante RUBEN DARÍO MORALES GARCÍA, en el que me desempeño como Director.

Cordialmente,

PhD. Jose Luis Villa

Director de Tesis

Nota de Aceptación

Firma del Presidente del Jurado

Firma del Jurado

Firma del Jurado

AGRADECIMIENTOS

El autor expresa especiales agradecimientos a:

PhD. Jose Luis Villa, por visualizar el potencial necesario para realizar y culminar este trabajo de grado, por su dedicación desinteresada en cada etapa de desarrollo del mismo.

Msc. Jorge Duque, por su valiosa colaboración en la obtención de algoritmos y sentencias de programación del protocolo Modbus RTU, e interés en la preparación de actividades de aprendizaje en temas de micro-controladores industriales.

Ing. Ramiro Caldera, por su invaluable y desmedida colaboración en programación y pruebas de los sistemas electrónicos diseñados.

BIOFILM S.A., por su valiosa ayuda en la adquisición de equipos para las pruebas, como también en el ofrecimiento del tiempo para terminación de este documento.

Tabla de Contenido

1. Resumen.....	14
2. Introducción	15
3. Estado de Arte.	19
3.1. Modelo y Estándares.	19
3.1.1. Modelo OSI.....	19
3.1.2. Estándar ISA /SP50.....	20
3.2. Modbus RTU.....	22
3.3. Sistemas de lógica reconfigurable FPGA para Protocolo Modbus RTU.....	25
3.3.1. Generalidades.	25
3.3.2. FGA como procesador de comunicaciones.	26
3.4. Comunicación inalámbrica industrial.	28
3.4.1. Actualidad de la comunicación inalámbrica industrial.	29
3.4.1.1. ISA sociedad de Sistemas de Instrumentación y Automatización:	29
3.4.1.2. IEEE Instituto de Ingenieros Electricistas y Electrónicos:	33
3.4.1.3. FCC Comisión federal de comunicaciones:	34
4. Diseño y requerimientos del enlace de comunicación.	37
4.1. Consideraciones del diseño.....	37

4.1.1.	Ambiente Industria típico y las modulaciones más óptimas.	37
4.1.1.1.	Técnica de modulación FHSS.	39
4.2.	Requerimientos del diseño.	41
4.2.1.	Generalidades del aplicativo propuesto.	41
4.2.2.	Especificaciones relevantes de la tarjeta DE1 de Altera.	44
4.2.3.	Radio de comunicación con uso de FHSS.	47
5.	Implementación de Modbus sobre FPGA.	49
5.1.	Metodología.	49
5.1.1.	Establecimiento de las necesidades del proceso a automatizar.	50
5.1.2.	Configuración del Hardware en la tarjeta FPGA DE1.	51
5.1.2.1.	Configuración de la tarjeta DE1 bajo la herramienta SOPC.	53
5.1.2.1.	Aplicativo de esclavo Modbus RTU en Nios II, Programación ANSI C. ...	58
5.1.2.1.1.	Recepción de datos por la UART RS232.	59
5.1.2.1.2.	Sistema de chequeo de tramas CRC-16.	62
5.1.2.1.3.	Ejecución de las funciones de Modbus RTU.	63
5.1.2.1.4.	Temporizaciones entre caracteres y tramas.	69
5.1.2.1.5.	Aplicaciones implementadas de diagnóstico.	71
5.1.2.1.6.	Envío de respuesta a solicitud.	73
5.1.3.	Mapa de memorias de la implementación realizada.	74

5.1.3.1.	Mapa de memoria para la función 01	74
5.1.3.2.	Mapa de memoria de la función 02.	75
5.1.3.3.	Mapa de memoria de función 03.	76
5.1.3.4.	Mapa de Memoria de función 04.	76
5.1.3.5.	Mapa de memoria para función 05.	77
5.1.3.6.	Mapa de memoria para función 06.	77
5.2.	Configuración de Radios FHSS y enlace radio.	78
5.2.1.	Configuración de Radio FHSS.	79
5.2.2.	Enlace de comunicación radio.	80
5.2.2.1.	Estado Libre (Idle):	81
5.2.2.2.	Estado de transmisión:	82
5.2.2.3.	Estado Recepción:	83
6.	Pruebas y ensayos del sistema dentro de un ambiente industrial.	84
6.1.	Generalidades de la Prueba.	85
6.2.	Medición de recursos a tarjeta DE 1.	89
6.3.	Pruebas de comunicación a la implementación.	90
6.3.1.	Pruebas al interior de planta (variación de distancia).	92
6.3.2.	Pruebas al exterior de planta (variación de distancia).	93
6.3.3.	Prueba de estabilidad de comunicación.	94

7. Conclusiones y trabajos futuros.	96
8. Referencias	102
9. Anexos.....	106
9.1. Programación en NIOS II de Altera.	106

Lista de Figuras

Figura 1. Niveles OSI.....	20
Figura 2. Clasificación según uso de un enlace inalámbrico, extraído de ISA 100 [22].	31
Figura 3. Matriz de tratamiento de enlaces ante ambientes con interferencia. [22]	32
Figura 4. Bandas ISM sin licencia, especificadas por la FCC.	35
Figura 5. Otras bandas especificadas por la FCC, sin licencia.	35
Figura 6. Esquema básico de un transmisor y receptor bajo el uso FHSS[22]	40
Figura 7. Envío de datos bajo FHSS[22].....	41
Figura 8. Conexiones en la tarjeta DE1 [23].	45
Figura 9. Radios de comunicación dispuestos para la implementación.....	47
Figura 10. Pasos generales de implementación del esclavo Modbus RTU (1).....	49
Figura 11. Pasos generales de implementación del esclavo Modbus RTU (2).....	50
Figura 12. Configuración SOPC del sistema citado.	53
Figura 13. Configuración de Procesador NIOS II.....	54
Figura 14. Configuración de SDRAM Controller.	55
Figura 15. Configuración de la UART, RS232C.....	56
Figura 16. Configuración PIO de LEDs verdes.....	57
Figura 17. Configuración de módulos de tiempo, control de tramas.....	58
Figura 18. Algoritmo simplificado de implementado en FPGA para esclavo Modbus RTU .	59
Figura 19. Diagrama de flujo para la recepción, evaluación y envío de datos Modbus RTU (1)	61

Figura 20. Diagrama de flujo para la recepción, evaluación y envío de datos Modbus RTU (2)	62
Figura 21. Tiempos entre bytes y entre tramas, soportados[4].....	70
Figura 22. Indicaciones de diagnóstico disponibles en la DE1.	73
Figura 23. Suiches de configuración de radios.[2]	79
Figura 24. Configuración de Cableado de Radio a DE1 (FPGA)[2]	80
Figura 25. Esquemático de enlace radio.[2]	80
Figura 26. Estados de operación, radios FHSS[2].	81
Figura 27 .Proceso de transmisión mensajes [2].....	82
Figura 28. Motor AC de 400Kw con variación de velocidad por frecuencia.	86
Figura 29. Variador de velocidad AC.....	86
Figura 30. Set de motores de ventilación para horno.	87
Figura 31. Módulos de campo con comunicación industrial.	87
Figura 32. Resultado de Quartus, aplicativo Modbus RTU	89
Figura 33. Muestra de pantalla de software maestro para pruebas de comunicación, configurado para el desarrollo de las pruebas en "Protool Pro" de Siemens.	92
Figura 34. Comportamiento de aciertos en tramas en interior de planta.....	93
Figura 35. Comportamiento de aciertos en tramas en exterior de planta.	94

Lista de Tablas.

Tabla 1. Características de la capa física de Modbus RTU [11].	22
Tabla 2. Características de la capa de enlace de Modbus RTU [11].	23
Tabla 3. Características de la capa de aplicación de Modbus RTU [11].	23
Tabla 4. Fuentes de interferencias más comunes [20].	38
Tabla 5. Caso de la implementación.	42
Tabla 6. Tabla de requerimientos.	43
Tabla 7. Características resumidas de la DE 1.	44
Tabla 8. Características resumidas del radio XStream – PKG [2]	47
Tabla 9. Mapa General de Memoria	74
Tabla 10. Mapa de memoria para la función 01.....	75
Tabla 11. Mapa de memorias para la función 02.	75
Tabla 12. Mapa de memorias para la función 03.	76
Tabla 13. Mapa de memorias para la función 04.	76
Tabla 14. Mapa de memoria para la función 05.....	77
Tabla 15. Mapa de memoria para la función 06.....	77
Tabla 16. Designación para ubicación en diagrama de planta.....	88

1. Resumen

En este trabajo se aborda la implementación de una red Modbus RTU en una FPGA como plataforma experimental de comunicación inalámbrica probada en un ambiente industrial típico, donde es posible encontrar sitios con alta contaminación de ruido eléctrico como también lugares con alta sensibilidad a señales de tipo electromagnético.

Las pruebas y ensayos están enfocadas a determinar las ventajas de uso de esta topología (hardware y software) en ambientes hostiles, normales en una industria.

El sistema a construir está basado en la topología maestro-esclavo, el participante esclavo de la red es conformado por una tarjeta de lógica reconfigurable FPGA, previamente configuradas para comportarse según la necesidad, el maestro es un PLC o PC con software maestro Modbus RTU, la tarjeta FPGA y el maestro a su vez cuenta con un radio de comunicación con Espectro Ensanchado por Saltos en Frecuencia (FHSS) y cuenta con modulación por cambio de frecuencia (FSK).

Dicho sistema es instalado, configurado y probado en un ambiente industrial, en donde es posible establecer el grado de desempeño de este tipo de ejecuciones y redes con usos particulares en la industria.

Después de las experimentaciones fue posible establecer la relevancia del tipo de modulación según el uso e interferencias presentes en ambientes de operación, temas como el cuidado ante interferencias también son tratados en el presente documento.

2. Introducción

Se ha demostrado que las redes industriales son una herramienta muy eficaz en los procesos de automatización, reduciendo de esta forma los tiempos de puesta en marcha, modificaciones y mejoras a los sistemas automáticos [3], en la búsqueda de fortalecer aún más su campo de aplicación, este documento propone una implementación de una plataforma experimental de comunicaciones industriales bajo el uso de una FPGA¹ y protocolo Modbus RTU junto con un enlace inalámbrico a 900Mhz.

Uno de los protocolos de comunicación industrial más ampliamente conocido es precisamente Modbus RTU, desde su desarrollo en 1979, este ha ganado una gran acogida en la industria, comunicando equipos inteligentes; con entradas, salidas y registros de datos, el protocolo ha tomado un puesto especial en la rama de la automatización como una de las redes de fácil implementación, contando con un gran número de equipos e interlocutores compatibles.

Basados en el éxito del protocolo Modbus, se plantea su uso dentro de una FPGA, con el fin de conformar una comunicación con topología maestro – esclavo usando como medio de enlace un sistema inalámbrico de radios con modulación FHSS².

Protocolos como el Modbus han sido ya desarrollados bajo dispositivos como ASIC³, el cual podemos llamar como el competidor más cercano de las FPGA, que puede llegar a ser más eficiente y rápido para la ejecución del sistema mencionado, sin embargo la FPGA ofrece grandes ventajas experimentales, como puede ser lograr hacer modificaciones a su

¹ por sus siglas en Inglés Field Programmable Gate Array

² Saltos en el espectro de frecuencia,

³ Circuito Integrado para Aplicaciones Específicas

algoritmo en búsqueda de una mejora en su adaptación al aplicativo a disposición de diseñador o usuario. En [18] Patrick Salaün ha propuesto la FPGA como una solución de bajo costo la cual puede ser una excelente respuesta a la necesidad de automatismos de alta velocidad los cuales con un PLC pueden llegar a ser más costosos tanto en implementación como en mantenimiento mismo. Esta es una de las tantas razones que hacen de la FPGA una herramienta muy poderosa para el desarrollo de sistemas de comunicación y automatización de bajo costo.

Otra publicación de especial importancia, dada su similitud con este trabajo, es el elaborado por Duque en [1], el cual implementa en un micro-controlador el Protocolo Modbus RTU y pone en servicio el sistema en una aplicación real de la industria, la ejecución del sistema sirve de soporte inicial para este documento.

En los capítulos posteriores se expone la creación de una solución para la experimentación en comunicación industrial, con el uso de una FPGA de Altera DE1, la cual es configurada bajo una arquitectura de microprocesador y programada bajo el entorno de desarrollo NIOS II en Ansi C, y es puesta en servicio como una estación esclava Modbus RTU.

La implementación citada cuenta a su vez con comunicación inalámbrica, por medio de un radio a 900Mhz dentro de la banda ISM, la cual enlaza con un maestro Modbus usando funciones de lectura-escritura de bits y registros.

Este documento a su vez propone pruebas al sistema para analizar factores en la comunicación inalámbrica tales como Número de mensajes recibidos correctamente versus distancia; y Estabilidad de enlace inalámbrico, todos bajo entornos industriales.

Esto con la finalidad de establecer una solución exitosa de bajo costo con el uso del protocolo Modbus RTU y tarjetas lógicas reconfigurables.

Los resultados experimentales nos permiten mostrar las excelentes prestaciones que esta plataforma ofrece a las experimentaciones de enlace de comunicación industrial, gracias a su gran adaptabilidad a diferentes arquitecturas y modelos, propias de su naturaleza de reconfiguración y programación de la FPGA. De igual forma se expone otra ventaja y es la capacidad de la arquitectura de soportar algoritmos y/o programaciones adicionales para funciones de clientes las cuales pueden ir desde diagnósticos de comunicación especiales para usuarios como también automatismos y controles a procesos productivos, que permiten mejorar la adaptación de este sistema a muchas situaciones industriales típicas.

El principal objetivo de esta investigación es la Implementación de Protocolo Modbus RTU y su análisis bajo propagación inalámbrica en entornos industriales. Junto con la implementación deseamos dar solución a la pregunta de investigación de este proyecto: ¿Qué tan confiable y robusto es la comunicación inalámbrica con protocolo Modbus RTU en diversos entornos industriales?

Este documento está organizado de siguiente manera, en el capítulo tres se presenta un estado del arte centrado en los sistemas de comunicación industriales y en particular en implementaciones de protocolo de buses de campo. En capítulo cuatro se desarrollan los elementos de diseño y los requerimientos del enlace de comunicación. En el capítulo cinco se presenta el aporte más amplio de este trabajo con la implementación del protocolo Modbus RTU sobre la FPGA. En el capítulo seis se presenta las pruebas y ensayos del

sistema dentro de un ambiente industrial. Finalmente el capítulo siete resume las principales conclusiones y esboza algunos de los posibles trabajos futuros.

3. Estado de Arte.

Dada la creciente demanda de industria en obtener cada vez más información proveniente de las redes de comunicación en sistemas de automatización dispuestos en campo, se hace necesario establecer un número de reglas y estándares con el fin de garantizar una eficiente utilización de las arquitecturas y flujos de información, este ha sido uno de los trabajos liderados por algunas organizaciones tales como ISO e ISA entre otras.

A continuación no solo detallaremos un poco sobre dichas organizaciones sino también en otra clase de aportes en cuanto a arquitecturas de comunicación refiere, todo esto con el fin de estudiar sus contribuciones a lo que hoy se conoce como comunicación inalámbrica industrial.

3.1. Modelo y Estándares.

3.1.1. Modelo OSI.

El modelo OSI propone una estructura muy amplia la cual es adoptada por cada red en diferentes formas (ciertos niveles son adoptados y otros no).

El estándar final fue publicado en 1984, como ISO 7498 [5], donde se destacan una arquitectura de 7 niveles o capas. Los cuales suponen una comunicación entre niveles y entre locutores.

El Sistema de niveles o capas OSI busca especificar el sistema de transmisión, el método de acceso a la red y todo lo referente a cómo realizar un intercambio de información eficiente entre dos o más interlocutores. Para el caso del sistema a implementar con

protocolo Modbus RTU donde solo se usan 3 de los 7 niveles propuestos, como se muestra en la figura 1.

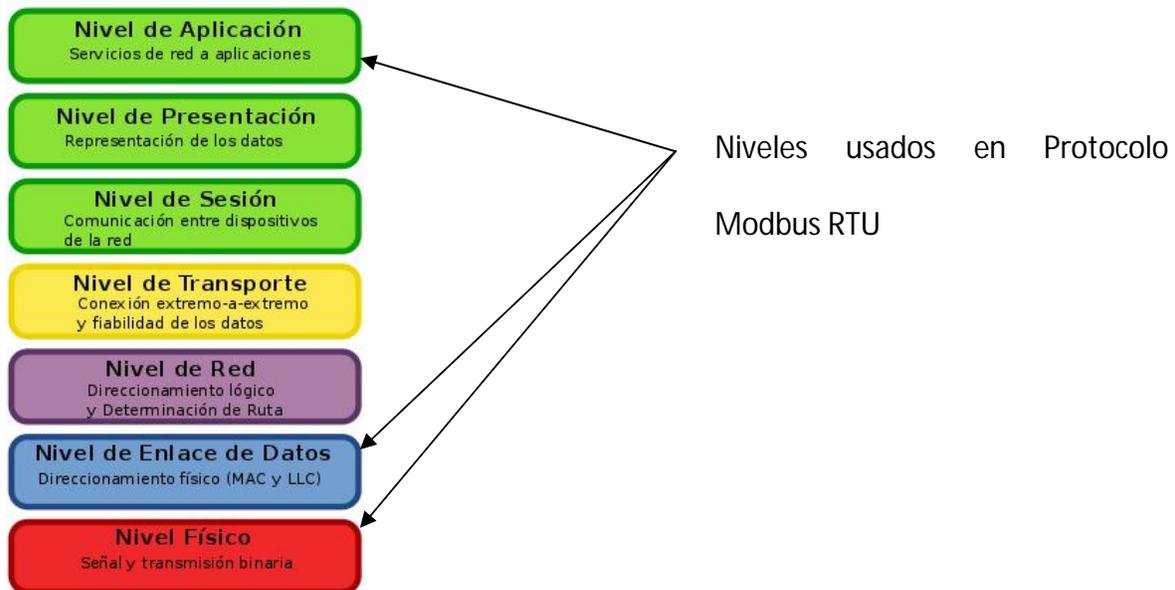


Figura 1. Niveles OSI.

3.1.2. Estándar ISA /SP50.

A pesar de que el modelo OSI es muy completo, casi todos los buses de campo ofrecen grandes problemas para poder satisfacer los niveles intermedios, del 3 al 6, dado que los fabricantes no previeron la conexión con otros buses.

Para completar el paquete de protocolo propuesto por el modelo OSI, la sociedad para instrumentación, sistemas y automatización, ISA, propone una serie de complementos o mejoras bajo la denominación ISA-SP50 [6], donde se trata de desarrollar las normas necesarias para definir las características que deben cumplir las señales (analógicas y digitales) usadas en medidas de proceso y control, y transmitir la información entre subsistemas o elementos separados de sistemas.

ANSI/ISA-50.1-1982 (Rev:1992) [6], como complemento al modelo OSI, este estándar propone 2 capas adicionales:

- Capa de usuario: Bloques de Funciones y gestiones de base de datos a disposición del usuario como tal para facilitar el control y adquisición de datos.
- Capa de Supervisión: funciones de monitoreo, parametrización y configuración de dispositivos entre otras, son las herramientas para la gestión de redes y sistemas.

Adicionalmente propone adiciones y/o mejoras a la composición de las capas 1, 2 y 7.

La capa 1 de ISA 50.1, se adapta al nivel 1 del modelo OSI, tal como es definido en ISO 7498, con la excepción de que los limitadores de trama también hacen parte de esta capa, especificándolo en ANSI/ISA-S50.02, parte 2 [6].

En caso de la capa 2, esta es mejorada complementándola con características de servicios de enlace de datos conveniente para comunicaciones críticas, el estándar que lo define es el ANSI/ISA-S50.03 [6], y por último la capa 7 de Aplicación, complementada por el estándar ANSI/ISA-S50.02, parte 5 y parte 6 [6], define una interface común para la interconexión de componentes de sistemas de medida y control.

ISA será parte importante para la implementación de la plataforma experimental.

3.2. Modbus RTU

Protocolo desarrollado por Modicon en 1979, diseñado para la comunicación con dispositivos inteligentes, como controladores, y unidades de adquisición de datos. Desde entonces se ha posicionado como uno de los protocolos más usados en la industria, la modalidad RTU inicialmente es ideal para la monitorización remota vía radio de elementos de campo [3], de allí su nombre (RTU, remote terminal Unit).

Dicho protocolo cumple con los modelos anteriores ISO e ISA en solo 3 de las 7 capas posibles: capa 1: físico, capa 2: enlace y capa 7: aplicación.

Modbus RTU ha tenido amplia utilización en el medio Industrial, soportado por una amplia gama de fabricantes y entidades que lo promueven y estudian. La organización Modbus – IDA constituida en el 2002 expone un amplio marco de aplicaciones para esta clase de protocolos facilitando su uso e implementación.

El protocolo Modbus solo usa 3 capas del modelo OSI. En las tablas 1, 2 y 3 se presentan algunas de las características de sus capas:

Capa OSI	Estándar	Conmutable con Potencia	Rango Velocidad	Máxima distancia
Física	IEEE 1451.2, TIA-485	NO	1.2Kbts – 12Mbts	1500m

Tabla 1. Características de la capa física de Modbus RTU [11].

Capa OSI	Estándar	Tipo de enlace	Detección de errores
Enlace	Ninguno	Maestro-esclavo con direccionamiento	1 bit, CRC-16

Tabla 2. Características de la capa de enlace de Modbus RTU [11].

Capa OSI	Estándar	Transferencia de datos.	Velocidad de adquisición
Aplicación	IEC61158	Registros	Dependiente del aplicativo

Tabla 3. Características de la capa de aplicación de Modbus RTU [11].

Modbus usa una topología Maestro-Esclavo, donde la comunicación es manejada solo por el maestro mediante el envío y recepción de mensajes a solicitud del aplicativo.

Bajo el funcionamiento de solicitud-respuesta se ofrecen servicios especificados en los códigos de función (método en el cual se solicitan acciones a cliente desde un servidor). Los códigos de función MODBUS son elementos de la solicitud, claramente visibles en la trama de solicitud y respuesta.

Existen dos variantes del protocolo Modbus: ASCII y RTU; siendo este último especialmente escogido para las aplicaciones industriales.

La modalidad RTU, nace de la aplicación de esta en comunicaciones con estaciones remotas, de allí deriva su nombre. Este modo es eficiente en cuanto ocupa por menor tiempo el medio en transmisión y recepción.

La modalidad ASCII básicamente usa caracteres ASCII para la codificación de sus caracteres junto caracteres espaciales para el final de trama permitiendo al dispositivo conocer con certeza el final de esta, con esto se evita hacer vigilancia a las tramas con el uso de temporizadores. Esta modalidad es menos eficiente que la RTU.

Para esta implementación se ha escogido la modalidad RTU, la cual ha sido ampliamente adoptada en el medio industrial.

En la actualidad existen varias formas de implementación de este protocolo:

- Sobre TCP / IP en Ethernet, en el cual básicamente los mensajes Modbus son encapsulados en los datos útiles de una comunicación TPC / IP. Este caso no será tratado en este documento.
- Comunicación asíncrona serie, transmisión a través de varios tipos de interfaces y medios, bajo las normativas: EIA/TIA-232-E, EIA-422, EIA/TIA-485-A, fibra, radio, etc).

Para el caso de esta implementación se ha adoptado la Modalidad RTU, bajo la transmisión serie usando EIA/TIA-232-E.

Modbus RTU cuenta con las siguientes ventajas:

- Es un protocolo abierto con mucha información disponible en diferentes fuentes bibliográficas [4], [7].

- Dada la amplia información disponible su implementación se facilita en gran gama de micro-controladores y procesadores.
- Se encuentra disponibles variados fabricantes⁴ que respetan sus reglas tanto en los diferentes medios de propagación como en la forma de manejar los datos y variables;

Las anteriores ventajas son algunas de las que dispone Modbus por lo cual lo hace un excelente candidato para ser usado en desarrollos de aplicaciones industriales, y más aun en esta implementación inicial la cual tiene como uno de los productos de investigación la ejecución de una plataforma experimental de comunicaciones inalámbricas industriales. En el capítulo de diseño e implementación se ofrecen más detalles sobre los diferentes aspectos del protocolo, también se puede disponer de la referencia [27] "MODBUS Over serial line specification and implementation guide V1.02".

3.3. Sistemas de lógica reconfigurable FPGA para Protocolo Modbus RTU.

3.3.1. Generalidades.

Una FPGA⁵ es un dispositivo semiconductor que contiene componentes lógicos programables e interconectables entre ellos de forma programada. Los componentes lógicos pueden ser re-programados para realizar funciones sencillas tales como: AND, OR,

⁴ Fuente: www.modbus.org, <http://www.modbus.org/companies.php#463>, Modbus Supplier Directory, 19/02/2010.

⁵Field programmable gate array.

XOR, NOT entre otras, o en conjunto para satisfacer funciones combinacionales mucho más complejas.

Desde su invención a mediados de los 80s por Ross Fresman, se ha vuelto muy útil para los desarrollos de arquitecturas, su gran ventaja es la posibilidad de permitir re-configurarle muchas veces sin volver a invertir otra vez en el mismo dispositivo, hoy en día una FPGA es tan versátil y poderosa que permite múltiples desarrollos en control y automatización.

Su principal competidor es el ASIC [18], el cual para la misma función y aplicación es capaz de ser más rápida y eficiente en consumo de energía, sin embargo las prestaciones de re-programación hacen de la FPGA un equipo de mucha utilidad para investigación, desarrollo, corrección de errores y pruebas. Adicionalmente es posible integrar mas aplicaciones a la función inicial ya sea para sugerir cambios al diseño original o para adicionar habilidades especiales los cuales pueden ser resueltos con el mismo integrado [8].

La anterior síntesis son algunas de las razones por las cuales hemos escogido la FPGA como herramienta para la implementación del protocolo Modbus RTU.

3.3.2. FGA como procesador de comunicaciones.

Recordando una de las razones por la cual ha sido escogida la FPGA para este desarrollo, la cual es la disposición de este a recibir funciones adicionales o complementarias a la intención inicial del diseñador, lo cual lo adaptan de mejor manera que los ASIC a procesos particulares.

Las FPGA han sido usadas en múltiples desarrollos que varían desde aplicaciones en simulación a proyectos aeronáuticos y militares, por lo tanto se alcanza tener mucha literatura sobre estas aplicaciones específicas, sin embargo, aplicaciones directas en las que se pueda contener el protocolo Modbus RTU y algunas funciones complementarias en la industria sobre la misma FPGA no se ha escrito mucho.

Con el fin de desarrollar arquitecturas de automatización con el uso de dispositivos FPGA, algunos autores como Laurence Picci y Patrick Salaün [8], han propuesto una alternativa de bajo costo con la implementación de sistemas basados en FPGA como una solución adicional para la modernización de sistemas de control instrumentados; dentro de las ventajas expuestas, presentan la baja inversión en dinero de los componentes y la dinámica de crecimiento tecnológico casi independiente del proveedor de equipos (gracias a la flexibilidad de desarrollo de la FPGA con la que el usuario final cuenta). El artículo se centra en especial sobre una ejecución en una planta nuclear en Francia, pero su punto de aplicación se puede generalizar con facilidad en muchos otros estudios de la ingeniería aplicada.

Por otro lado algunos autores han expuesto y desarrollado el protocolo Modbus RTU con el uso ya sea de micro-controladores, es el caso del trabajo desarrollado por el Duque [1], el cual implementó el protocolo Modbus-RTU esclavo en el sistema de desarrollo Adapt9s12E128, en él mostró sus ventajas y rendimiento costo-beneficio de su implementación y puesta en servicio. Además preparó un camino futuro para dos posibles trabajos posteriores:

- Implementación del protocolo Modbus TCP/IP en micro controladores, capaz de tener una conexión con Internet para ser accedida.
- Utilizar el protocolo Modbus para el desarrollo de aplicaciones de monitoreo y control a nivel industrial,

Es en este último donde esta implementación fundamenta su objetivo, en especial sobre el monitoreo de procesos industriales bajo el uso del protocolo Modbus RTU. El documento [1] ha servido como guía para la preparación y desarrollo de este material.

De igual manera ZUO Hong-fei, JIE Xin-chun, WANG Zhi-chun [17], han presentado un trabajo sobre la implementación del protocolo Modbus RTU sobre una FPGA, probada sobre un ambiente industrial, mostrando sus ventajas en múltiples aplicaciones, por desgracia dicho trabajo no se ha podido encontrar en otras fuentes diferentes a las asiáticas, por lo que no ha podido ser muy socializado en esta parte del mundo.

En fin, se ha escrito de forma sustancial sobre el protocolo en micro-controladores, pero en realidad es muy poco lo que ha sido aplicado o implementado en dispositivos como FPGA en función del protocolo Modbus RTU para aplicativos propios de la industria.

3.4. Comunicación inalámbrica industrial.

Hablar de comunicación inalámbrica vincula muchos tipos de redes que en la actualidad usamos, desde la señal de televisión o radio convencional hasta señales digitales para la navegación en internet, para el caso en que nos centramos en este documento

revisaremos solo los temas pertinentes a la evolución de las redes inalámbricas en cuanto a la industria refiere.

A continuación se expondrá la actualidad de las comunicaciones inalámbricas a la luz de las entidades que más se esfuerzan por su desarrollo.

3.4.1. Actualidad de la comunicación inalámbrica industrial.

Desde que fueron reservadas las bandas no comerciales ISM definidas por la FCC, estas han tenido un espacio para la expansión y crecimiento, se ha iniciado con comunicaciones RF con solo modulación FSK, pero ahora ya se cuenta con saltos en frecuencia FHSS, secuencia directa DSSS como también divisiones de frecuencia ortogonales OFDM.

Muchas entidades han demostrado mucho interés en las comunicaciones inalámbricas para la Industria como es el caso de ISA, IEEE y FCC, a continuación más detalle:

3.4.1.1. ISA sociedad de Sistemas de Instrumentación y Automatización:

En su publicación ISA - 100.11a – 2009: “Sistemas inalámbricos para la automatización industrial: Control de procesos y aplicaciones relacionadas” [24], este estándar trata sobre sistemas de comunicación inalámbrico confiable y seguro para el monitoreo no críticos, alertas, supervisión, operaciones de lazo abierto y aplicaciones de control en lazo cerrado. De igual forma define el conjunto de protocolos, administración del sistema y las especificaciones de seguridad para la conectividad inalámbrica de baja velocidad de datos

con dispositivos fijos, portátiles y móviles de soporte de los requerimientos de consumo de un poder limitado.

Con el fin de satisfacer las necesidades de los usuarios inalámbricos industriales y los operadores, este estándar también proporciona una plataforma de robustez a sistemas inalámbricos en presencia de interferencias propias de entornos industriales y otros sistemas inalámbricos, como pueden ser los que respetan las normas IEEE 802.11 x, IEEE 802.15 x, IEEE 802.16 x entre otras [24].

En especial la norma hace énfasis en las capas de conexión, de red, de transporte y de aplicación, como también trata el sistemas de conexión entre redes denominados "Gateway".

ISA en su edición TR100.00.01—2006 [22] también realiza dos grandes aporte a la clasificación de la clase de enlaces inalámbricos industriales, usando como factores tales como criticidad y niveles potenciales de interferencia.

- Clasificación en función de la criticidad en su uso.

Se establecen 6 categorías para diferentes tipos de enlace, que van desde Clase 0: sistemas de seguridad con alta criticidad a Clase 5: sistemas de monitoreo a advertencias de tipo no crítico. Ver la figura 2.



Usage classes of wireless data networks

Safety	Class 0 : Emergency action <i>(always critical)</i>	↑ Importance of message timeliness increases
Control	Class 1: Closed loop regulatory control <i>(often critical)</i>	
	Class 2: Closed loop supervisory control <i>(usually non-critical)</i>	
	Class 3: Open loop control <i>(human in the loop)</i>	
<small>NOTE: Batch levels* 3 & 4 could be class 2, class 1 or even class 0, depending on function *Batch levels as defined by ISA S88, where L3 = "unit" and L4 = "process cell"</small>		
Monitoring	Class 4: Alerting <i>Short-term operational consequence (e.g., event-based maintenance)</i>	
	Class 5: Logging & downloading/uploading <i>No immediate operational consequence (e.g., history collection, SOE, preventive maintenance)</i>	

- **Alarms**
 - Any class *(human or automated action)*
- **Wireless worker**
 - Class 3 – 5 *(access is usually proxied)*

Figura 2. Clasificación según uso de un enlace inalámbrico, extraído de ISA 100 [22].

- Tratamiento según niveles de interferencia.

Este aporte desea poner en claro cuando es necesario el arreglar de mejor manera el enlace previamente clasificado con el fin de mejorar su inmunidad, es decir para un enlace de un control en lazo cerrado, el cual puede llegar a ser crítico con clasificación 1, dentro de un ambiente con bajos niveles de interferencia no necesita ser modificado, en cambio el mismo sistema en un ambiente muy contaminado precisa mucho más cuidado y más medidas de mejora.

		RF Environment - Actual or Potential Interference		
		Low	Moderate	Heavy (or Unpredictable)
Safety	0	Warrants Concern	Warrants Concern	Warrants Concern
	1	Needs Care	Warrants Concern	Warrants Concern
Control	2	Needs Care	Needs Care	Warrants Concern
	3	Needs Care	Needs Care	Needs Care
Monitoring	4	Benign	Needs Care	Needs Care
	5	Benign	Benign	Needs Care

Legend:

Benign	Needs Care	Warrants Concern
--------	------------	------------------

Figura 3. Matriz de tratamiento de enlaces ante ambientes con interferencia. [22]

Con estos aportes ISA desea mostrar la importancia que tienen ciertos tipos de enlaces y sugiere tener en cuenta posibles cambios y/o revisiones al sistema, si las condiciones ambientales tales como interferencias así lo necesitan, esto en busca de mantener de forma sostenida la calidad del enlace y las prestaciones propias de su uso.

3.4.1.2. IEEE Instituto de Ingenieros Electricistas y Electrónicos:

El instituto dedicado entre otras cosas a la estandarización y creación de normas, ha desarrollado normas referentes a comunicación IEEE 802, que van desde la IEEE802.1., hasta la IEEE802.22.

Donde nuestro especial interés se centra en las norma IEEE802.11, dado que compartiríamos el medio y mucha relación con el tipo de redes que este expresa.

La norma se dedica a la transmisión de datos inalámbricos, con varias adiciones y/o extensiones desde que se originó. El concepto inicial fue especificar las velocidades de transmisión de 1 y 2 Megabits para una señal Infrarroja.

Una de las últimas publicaciones es la IEEE802.11y publicada en 2008, la cual permite comunicaciones entre los 3.6 y 3.7Ghz en ondas de radio.

A continuación las más empleadas en la actualidad para comunicaciones inalámbricas:

- IEEE 802.11b: estándar de 11Mbits en la Banda IMS de 2.4Ghz.
- IEEE802.11a: es la evolución del estándar IEEE802.11b, a velocidad de 54Mbits y 5,7Ghz.
- IEEE802.11e: Añade calidad de servicio al estándar IEEE802.11.
- IEEE802.11f: Añade a estándar IEEE802.11 factores de movilidad.
- IEEE802.11g: Equivalente al a, pero en la banda 2.4Ghz.

- IEEE802.11h: Intenta mejorar la potencia de transmisión.

La norma aun está siendo extendida, para el 2010 o 2011 se espera la IEEE802.11v.

En conclusión casi todas las normas establecidas de la IEEE para la comunicación inalámbrica están enmarcadas en las comunicaciones bajo el protocolo Ethernet y muchas de estas comunicaciones están situadas en las bandas de los 2.4Ghz y 5.4Ghz. Para el caso de nuestro desarrollo la frecuencia de trabajo es la cercana a la 900Mhz, por lo cual no tenemos mucha interacción con estas bandas, sin embargo hacen parte de la tendencia más importante en transmisión de datos ya disponible en el mercado para muchas de las aplicaciones.

3.4.1.3. FCC Comisión federal de comunicaciones:

Unos de los grandes aportes de la FCC en comunicaciones inalámbricas obedece a la especificación de las bandas sin licencia, en especial la banda ISM, la cual es de uso público para muchas aplicaciones en la industria, bajo esta última está contemplada nuestra aplicación. La figura 4 resume las bandas ISM especificadas por la FCC.

Banda	Frecuencia Gama (MHz)	Radio Requisito	Otros Requisitos
ISM	900–928	Espectro ensanchado FCC 15.247 EE.UU.	dBm 36; limitada disponibilidad, Estados Unidos, Canadá, Australia y partes de América del sur
ISM	2400–2483.5	Espectro ensanchado 15.247 De la FCC en Estados Unidos; ETS 300 328 en EU	36 dBm para punto - a-multipunto en Estados Unidos; 3-por-1 regla para pt a pt en Estados Unidos; 20 limite de Pire de dBm para UE
ISM	5725–5850	Espectro ensanchado FCC 15.247 EE.UU.	dBm 36 Pire

Figura 4. Bandas ISM sin licencia, especificadas por la FCC.

Adicionando así un requisito para la comunicación de ISM, que es el espectro ensanchado, el cual respetamos y utilizamos en nuestro enlace.

Adicionalmente la FCC ha especificado otras bandas sin licencia, la cuales ya no toman el nombre de ISM, estas se resumen en la figura 5.

Banda	Frecuencia Gama (MHz)	Radio Requisito	Otros Requisitos
U-NII / HiperLAN2	5150–5250	FCC 15.401 LOS ESTADOS UNIDOS; UE TS 101 475	dBm 23 Pire; interior sólo para uso
U-NII	5250–5350	FCC 15.401	30 dBm Pire
U-NII	5750–5825	FCC 15.401	dBm 36 Pire

Figura 5. Otras bandas especificadas por la FCC, sin licencia.

La norma FCC 15.247 publicada el 23 de julio de 1996, formaliza el uso de las bandas ISM bajo el uso de ciertas reglas, como los decibles máximos permitidos, la cantidad de

canales mínimos exigidos, el tiempo de residencia en cada canal (máximo y mínimo), entre otras.

La anterior norma es la respetada en nuestro aplicativo, dado que usamos la banda ISM en el rango de los 900Mhz, para la transmisión de datos.

4. Diseño y requerimientos del enlace de comunicación.

En este capítulo no solo discutiremos como se llevó a cabo la implementación del protocolo en la tarjeta de desarrollo FPGA DE1 de Altera y también algunas consideraciones previas a la implementación las cuales tienen mucha importancia en el desarrollo de la plataforma experimental.

4.1. Consideraciones del diseño.

4.1.1. Ambiente Industria típico y las modulaciones más óptimas.

El ambiente en el cual se desempeña una comunicación inalámbrica tiene un papel muy importante en la confiabilidad del enlace, de allí que se debe estudiar más en detalle las circunstancias que lo rodean y así establecer el grado de confiabilidad de dicho enlace.

El ambiente industrial moderno está cubierto por múltiples señales que pueden influir negativamente en las comunicaciones a nivel inalámbrico, motores de media y alta potencia, variadores de velocidad, equipos electrónicos de conmutación y otros elementos modernos, todos estos son fuentes de ruido e interferencias relevantes.

Las señales de interferencia en la industria se pueden clasificar en dos categorías diferentes, banda ancha y banda estrecha. Las señales de interferencia de banda ancha tienen un espectro de energía constante en muchas de las frecuencias propias, usando

alta energía. Por lo general son emitidas por error por sus fuentes, en cambio las señales de interferencia de banda estrecha son emisiones intencionales y tienen mucho menos energía. Ambos interferencias tienen diferentes tipos de efecto en la degradación de la confiabilidad del enlace inalámbrico [20].

En la tabla 4 se enuncian algunos tipos de fuentes de esta clase de ruidos o interferencias más comunes.

Interferencias de Banda Ancha	Interferencias de Banda Angosta
Motores	Teléfonos celulares.
Inversores, Controladores de Silicio y rectificadores, circuitos con SCR's	Radio y Transmisores de Televisión.
Reguladores de voltaje, sistemas de encendido de llama, generador de pulsos.	Generadores de señales, Sistemas de UPS, Hornos Micro-Ondas
Equipos de soldadura	Luces y balastos electrónicos.
Convertidores de frecuencia o variadores de Velocidad DC y AC	Sistemas con Microprocesadores, transmisores de señal ISM, generadores industriales de alta frecuencia.

Tabla 4. Fuentes de interferencias más comunes [20].

Con el fin de hacer el sistema de comunicación inalámbrica más confiable ante la presencia de alguno de estos tipos de ruidos, se dispone de las opciones de modulación en

el espectro, los cuales reducen significativamente la influencia negativa de dichas interferencias.

Las técnicas más comunes de modulación son: Espectro ensanchado por secuencia directa (DSSS⁶) y espectro ensanchado por saltos de frecuencia (FHSS⁷), la escogencia del tipo de técnica de modulación depende del ambiente y aplicación de la misma [20].

La técnica de modulación DSSS, es ideal para sistemas de alta velocidad pero solo es aplicable a ambientes con moderados niveles de interferencia, en cambio la técnica FHSS es aplicable a ambientes con alto grado de interferencias, garantizando niveles de seguridad más altos dado el poco tiempo de permanencia en una misma frecuencia, pero disminuye significativamente las velocidades de transmisión de los datos.

Para el caso de nuestra experimentación se ha considerado hacer pruebas en ambientes con media y alta contaminación, de manera paralela se contempla el uso del protocolo Modbus RTU el cual no exige velocidades altas, por lo cual es apropiado usar la técnica de modulación FHSS.

A continuación algunas descripciones útiles de la técnica de modulación FHSS.

4.1.1.1. Técnica de modulación FHSS.

En la modulación FHSS los datos son divididos en paquetes de información, dichos paquetes son enviados a través de varias frecuencias, esto es conocido como "Hopping

⁶ En esta técnica se generan bits redundantes que componen la señal.

⁷ Envíos momentáneos de tramas en diferentes frecuencias por periodos muy cortos.

Pattern", la intención de enviar la información por varias frecuencias obedece a fortalecer la seguridad y la inmunidad ante interferencias, ya que si la información fuera enviada por una sola frecuencia sería muy fácil interceptarla. La figura 6 presenta el esquema básico de un transmisor y receptor bajo el uso FHSS.

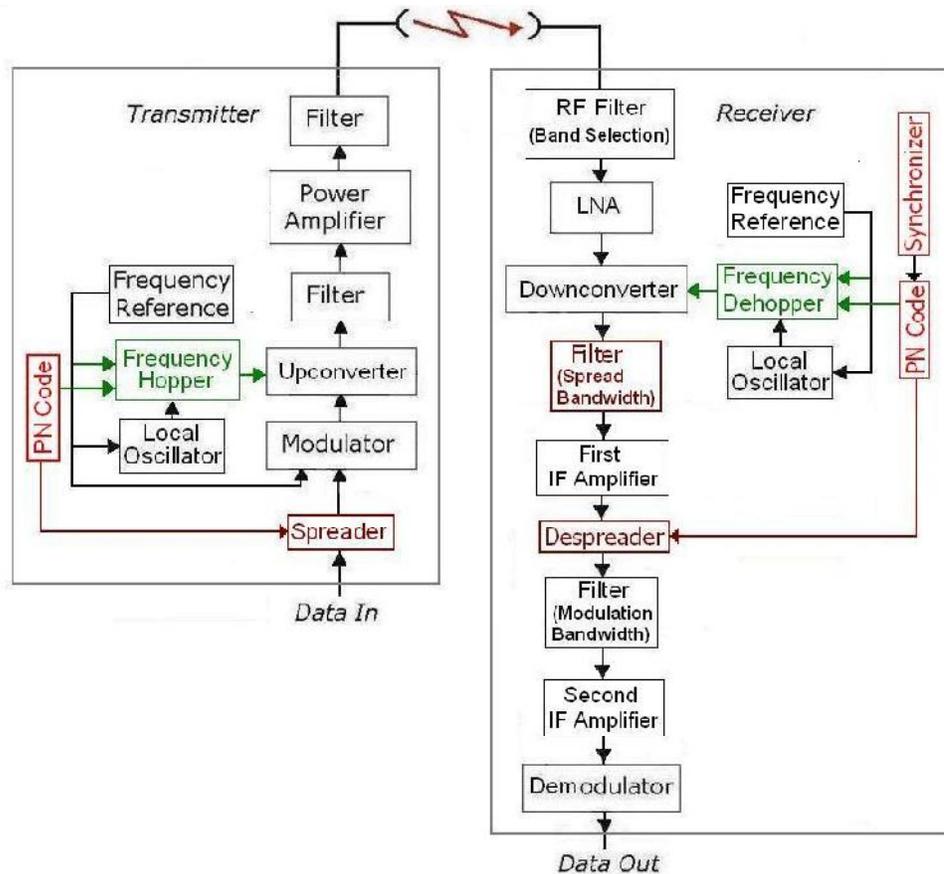


Figura 6. Esquema básico de un trasmisor y receptor bajo el uso FHSS[22]

La técnica FHSS depende en gran parte del buen desempeño de generador pseudoaleatorio, el cual determina cómo será el comportamiento de envío de datos en el espectro de frecuencias, tanto el emisor como el receptor deben ser conocedores de la misma información pseudoaleatoria, para el caso del receptor el uso del código debe estar sincronizado con el trasmisor. Dicho código generador está

conectado a un buffer de frecuencias, las cuales serán usadas para transmitir los datos en un pequeño instante de tiempo, tal como se puede esquematiza en la figura 7.

Frequency Hopping Spread Spectrum

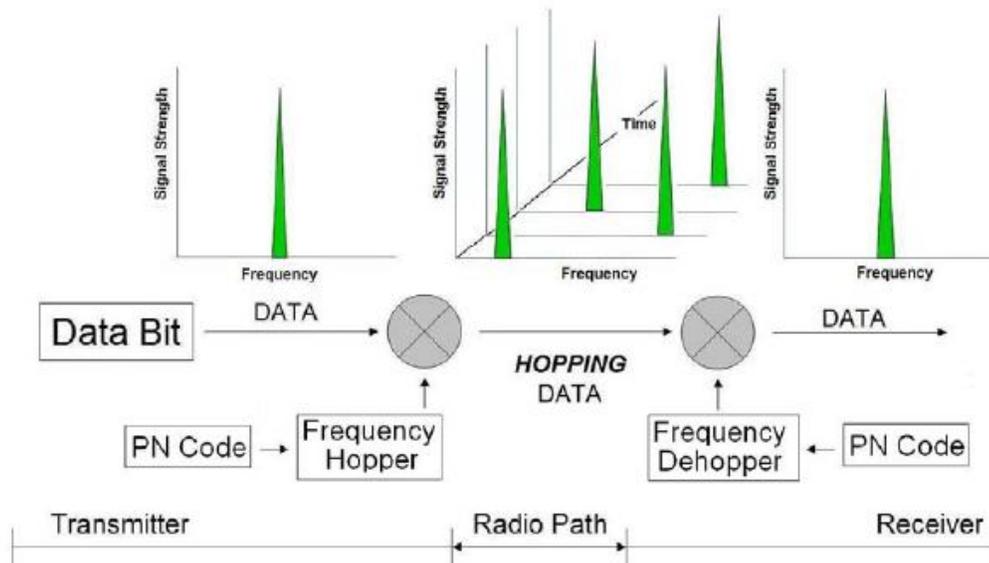


Figura 7. Envío de datos bajo FHSS[22]

4.2. Requerimientos del diseño.

4.2.1. Generalidades del aplicativo propuesto.

Como la idea principal de esta implementación es formar una plataforma experimental para el desarrollo y estudio de las redes de comunicación Industrial con enlaces inalámbricos, se plantean algunos requerimientos mínimos.

Se desea establecer algunos parámetros previos, los cuales deben ser satisfechos por esta primera parte del trabajo.

La tabla 5 muestra los requerimientos formales, usando la metodología propuesta por [20].

<p>Narración de la aplicación: Monitorea variables digitales remotas de sensorica dispuesta en campo, como la activación de salidas a dispositivos actuadores, los cuales están dispuestos en lugares donde no es muy viable el trazado de una red o cableado de comunicación. Por medio de una comunicación inalámbrica leer - escribir sobre variables en campo y hacer llegar estos datos a un sistema de visualización y monitoreo. Se propone el protocolo Modbus para este aplicativo.</p>
<p>Caso: Lectura_escritura_variables_digitales_inalámbricamente.</p>
<p>Sistemas:</p> <ul style="list-style-type: none"> · Sistema de Supervisión Modbus RTU. · Enlace inalámbrico · Tarjetas FPGA Esclavas Modbus.
<p>Pre-condiciones: Condición de planta con ruido electromagnético, de banda ancha y banda angosta de forma moderada.</p>
<p>Pasos:</p> <ol style="list-style-type: none"> 1. Configuración de la estación esclava. <ol style="list-style-type: none"> 1.1. Configuración del Arquitectura en Hardware de la FPGA DE1 1.2. Configuración de Software NIOS II en Tarjeta DE1. 1.3. Programación de Sentencias en Ansi C <ol style="list-style-type: none"> 1.3.1. Programación de sentencias adicionales de diagnostico 1.4. Pruebas y ensayo por puerto Serial RS232. 2. Configuración de radios <ol style="list-style-type: none"> 2.1. Configuración de las interfases RS232 de los radios. 2.2. Revisión de las condiciones de comunicación con respecto al medio. 2.3. Pruebas y mediciones de tramas exitosas según la disposición a listada. 3. Configuración del sistema maestro Modbus RTU. <ol style="list-style-type: none"> 3.1. Enumerar y configurar las variables de lectura y escritura. 3.2. Esquematización y colocación de variables en puntos del mímico. 3.3. Configuración del puerto y interfase lógica de Modbus RTU. 3.4. Pruebas y puesta en marcha del sistema.
<p>Variaciones:</p>
<p>Requerimientos adicionales:</p> <ol style="list-style-type: none"> 1. Tableros para la disposición de las tarjetas FPGA. 2. Sujeción de radios, en lo posible condicionando línea de vista. 3. Cableado de sensores y conversores de señal a tarjeta FPGA. 4. Cableado de alimentación a tarjeta esclava.
<p>Comentarios adicionales:</p>

Tabla 5. Caso de la implementación.

Adicionalmente es necesario establecer los requerimientos de comunicación, los cuales se resumen en la tabla 6, esto con el fin de conocer todas las necesidades y limitaciones previas a su implementación [20].

Requerimientos	Valor verbal	Valor numérico	Observación
Tipo de datos: Datos de señales de entrada y salida digitales (on-off)	Velocidad de actualización lenta	Lectura: < 3seg Escritura: < 5Seg	Simulación de señales, on-off de sensores
Tipo de control: Solo monitoreo y escritura de señales no críticas	Monitoreo en gran parte y escritura de solo pocas		Señales de comportamiento no delicadas
Velocidad comunicación:	Lento	19.200Kps	
Frecuencia Scan de Esclavo	Medio	<5 Seg max	
Tiempo para actualización de variables max ("Latency")	Medio	Entre 1 seg a 5 seg.	
Numero de Nodos	Pocos	2 a 3 estaciones	
Densidad de estaciones por m2	Poco	>10m2	
Tipo de ambiente: <u>Alto:</u> Industrial contaminado, posibles daños al equipo, muchas señales de ruido, <u>Medio:</u> Industrial convencional, motores, poca señales inalámbricas, <u>Bajo:</u> Pruebas laboratorio.	Medio		Temperatura ambiente alta dada la posible cercanía con sistemas de calentamiento.
Tiempo de vida necesario:	Años		
Distancia de monitoreo	Metros	<200Metros	Línea de vista (200m)
Mobilidad:	Sistema fijo		
Seguridad de los datos	Normal, monitoreo		Lazo abierto, revisión de variables.
Banda de comunicación:	Banda permitida ISM	900Mhz	

Tabla 6. Tabla de requerimientos.

Las anteriores tablas muestran las condiciones de diseño respetadas por la implementación realizada, el cual como objetivo fundamental es la lectura remota de variables con ciertos requisitos de velocidad, dentro de un ambiente industrial con medio-alto ruido de equipos eléctricos.

A continuación se describirá la configuración de la FPGA y radios en pro de satisfacer las condiciones previamente mostradas.

4.2.2. Especificaciones relevantes de la tarjeta DE1 de Altera.

Se cuenta con una tarjeta de desarrollo DE1 FPGA ALTER con las siguientes características, resumidas en la tabla 7 y cuyo diagrama se encuentra en la figura 8:

Características resumidas de la DE1 de Altera [23]	
Microprocesador:	Cyclone II FPGA y dispositivo de configuración serial EPCS4.
Memorias:	8-MB SDRAM, 512-KB SRAM, 4-MB Flash
Relojes:	50Mhz, 27 Mhz, entrada para reloj externo.
Puertos:	USB, RS232, Entrada PS/2, Expansión GPIO 1 y GPIO 2, SD Card Socket.
Salidas:	Earphone, mic, VGA port, 10 LEDs Rojos, 8 LEDs Verdes, 7 display de segmentos
Entradas:	10 Toggle Switchs, 8 Pulsadores.

Tabla 7. Características resumidas de la DE 1.

Todas las señales, ya sean de entrada o salida, están siendo siempre controladas por la Cyclone II (FPGA), esto añade grandes prestaciones en diseño, dado que todo puede ser controlado y monitoreado desde la FPGA con el algoritmo que se le destine. En la figura 9 se muestra un esquemático del diseño propio de la tarjeta DE1 con Cyclone II [23].

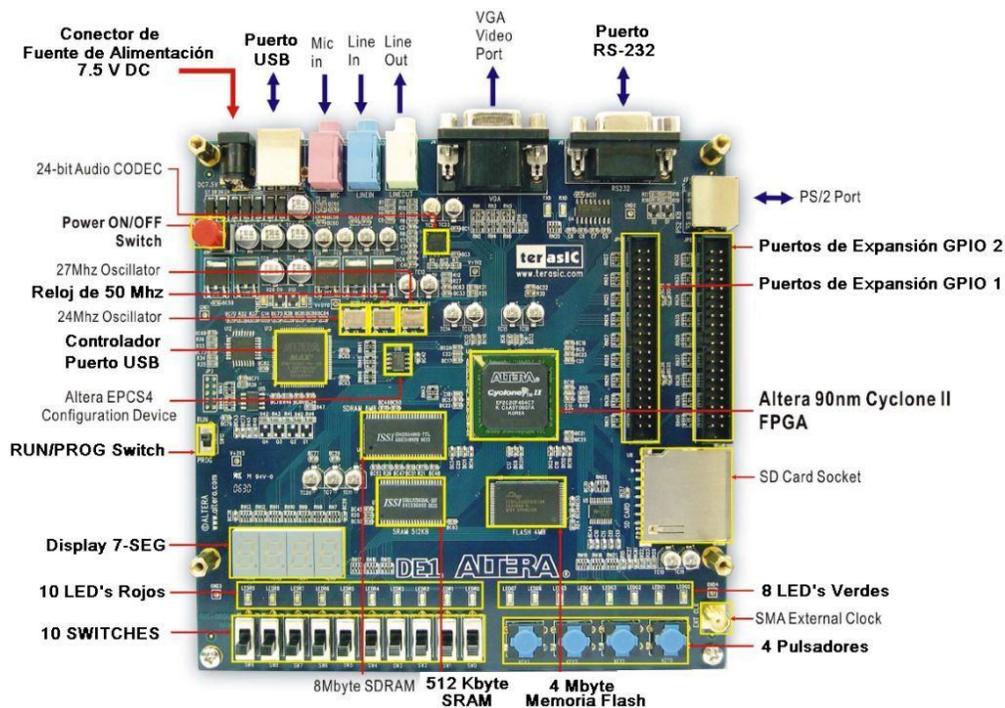
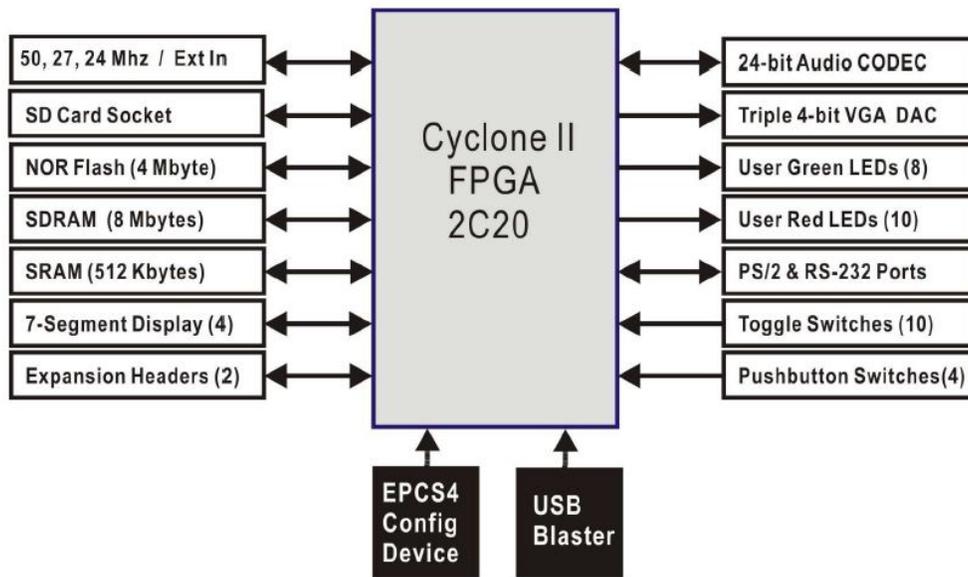


Figura 8. Conexiones en la tarjeta DE1 [23].

Las opciones y/o conexiones tales como SD card Socket, audio y video, no son usadas en esta implementación.

4.2.3. Radio de comunicación con uso de FHSS.

En búsqueda del cumplimiento de las especificaciones de la comunicación, establecida en las tablas anteriores, se escoge un radio de la marca XStream-PKG-R, los cuales garantizan comunicación de 450metros en interiores, con potencia de 100mW, en la banda ISM [2]. Las características de este radio se resumen en la tabla 8, y en la figura 9 se presenta una imagen del radio.

Características resumidas del radio XStream-PKG [2]	
Rango Comunicación:	450m @ 900Mhz.
Potencia de salida:	100mW(20dBm)
Sensibilidad recepción:	-107dBm@19200bps
Frecuencia de trabajo:	ISM 902-928Mhz
Velocidad de datos:	19200bps
Modulación:	FSK / FHSS
Regulación:	FCC part 15.247
Puertos:	RS232 y RS485 (configurable)
Temperatura max:	85°C

Tabla 8. Características resumidas del radio XStream – PKG [2]



Figura 9. Radios de comunicación dispuestos para la implementación.

Los radios trabajan en la banda frecuencias de 902 a 928MHZ, bajo el uso de FHSS, el cual les ofrece algo de ventaja en cuanto al enlace físico que estos pueden establecer. Además, disponen un puerto configurable RS-232 /RS485 el cual facilita la conexión con la tarjeta de DE1 de altera. Adicionalmente cumple con un buen rango de temperatura (condición establecida como requisito).

5. Implementación de Modbus sobre FPGA.

5.1. Metodología.

Basados en el cumplimiento de los requerimientos solicitados en las tablas 2 y 3, se procede a establecer los siguientes pasos, ver las figuras 10 y 11 con generalidades de cada paso:

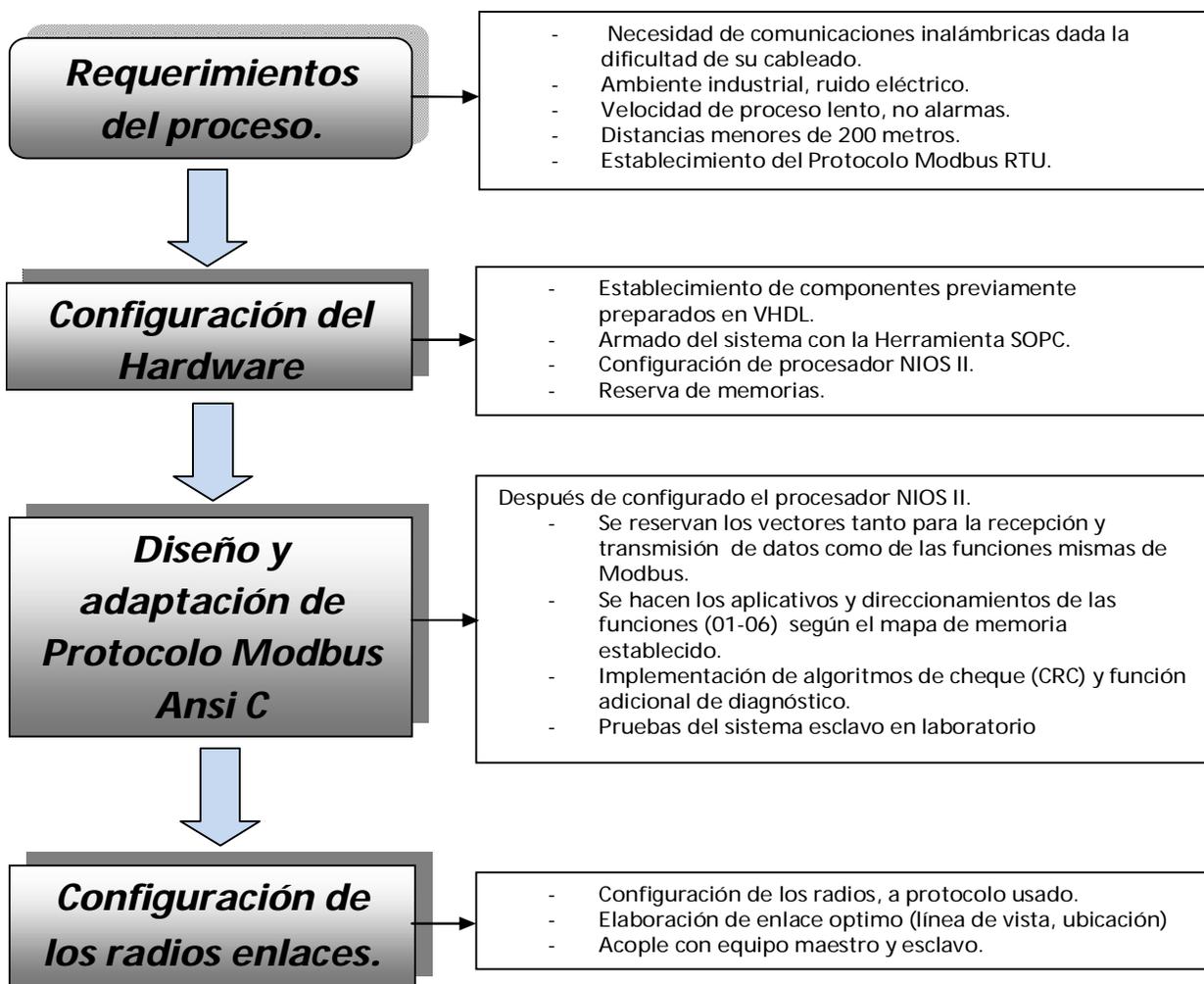


Figura 10. Pasos generales de implementación del esclavo Modbus RTU (1)

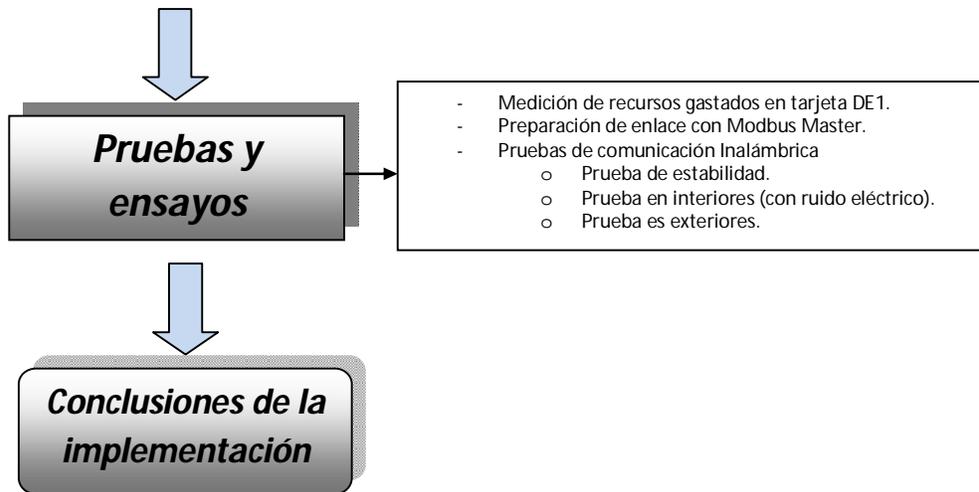


Figura 11. Pasos generales de implementación del esclavo Modbus RTU (2)

Cada paso es descrito anteriormente se encuentra más detalle a continuación:

5.1.1. Establecimiento de las necesidades del proceso a automatizar.

Basados en la necesidad de monitorear procesos con las dificultades para la obtener los valores de las variables por medios convencionales (cableado), se hace necesario en él pensar en una comunicación inalámbrica para dicho fin, por lo cual se establecen algunos requerimientos mínimos, tales como:

- Comunicación inalámbrica, robusta ante perturbaciones de ruido eléctrico.
- Distancias entre interlocutores que puede llegar hasta 200 metros de distancia.
- Variables de proceso, con propósitos de monitoreo (con posibilidad de escritura, sin alarmas).

Más información sobre las necesidades las cuales respeta esta implementación, se encuentran plasmadas en la tabla 4 y 5.

5.1.2. Configuración del Hardware en la tarjeta FPGA DE1.

El protocolo Modbus-RTU ha implementar en la tarjeta DE1, se logra bajo la programación del dispositivo con algoritmos en lenguaje ANSI C, junto con drivers de aplicación de los registros de configuración de la DE1 Altera FPGA con un procesador Nios II. Adicionalmente, se incorpora un microprocesador descrito en VHDL previamente, con un bus de datos de 32 bits.

Todos los periféricos del sistema, incluyendo Nios II están interconectados a través de un bus del sistema llamado "Avalon Switch Fabric".

La aplicación de ANSI C elaborada en ambiente NIOS se almacena en las memorias volátiles de la tarjeta DE1.

El microcontrolador ejecuta las instrucciones del programa de aplicación y se encarga de llevar a cabo la adquisición de datos de entrada (Cargar Trama de consulta Modbus-RTU), controla las salidas (Trama de respuesta Modbus-RTU) y cumplimiento de los requerimientos de la comunicación del protocolo tales como: Calculo de CRC y detección de errores.

Se realiza un programa respetando las convenciones establecidas por el algoritmo Modbus RTU [4], tales como reconocimiento de dirección esclava, tamaño de tramas,

algoritmos de chequeo, etc; recomendados por norma, para lograr la comunicación con un host o PC como equipo maestro de forma satisfactoria, ver anexo 1.

El programa debe cumplir con funciones tales como:

- Reserva vectores de entrada y salida de datos para ser usado por las funciones de comunicación, los cuales son dependientes de la estructura propia de la DE1 (tamaño de entradas y salidas dispuestas desde el hardware).
- Mantener una coherencia de tiempos los cuales podrán determinar los finales de trama y/o faltantes de caracteres de los mismos [4].
- Ejecutar la revisión de los datos tanto para conocer cuál es la función requerida como también calcular y comparar el CRC recibido.
- Elaborar una rápida respuesta a la solicitud inmediatamente recibida, sin incumplir con el tiempo máximo para responder a dicha solicitud.
- La Ejecución de una subrutina para los cálculos de datos estadísticos como es el caso de tramas recibidas exitosamente, ver ecuación 1, la ecuación ayuda a determinar en sitio de instalación el desempeño de la comunicación.

Dado que se cuenta con las prestaciones de un microcontrolador es posible contar con las opciones ilustradas en la sección 3.3.2., donde una de las más relevantes es la adaptabilidad de su sistema sin realizar cambios al hardware. Toda la configuración de

este se efectúa sobre Quartus II y SOPC⁸. Con este sistema embebido las aplicaciones funcionan en un orden más rápido que los microcontroladores convencionales.

5.1.2.1. Configuración de la tarjeta DE1 bajo la herramienta SOPC.

Junto a la herramienta SOPC se crea y configuran los diferentes módulos que conformarán el sistema dotado de un microcontrolador, puertos, memorias, salidas, entradas, comunicación y demás elementos necesarios para la estructura, todos dependientes de una misma señal de reloj e interconectados por un bus denominado "Avalon switch fabric"[25], ver figura 12.

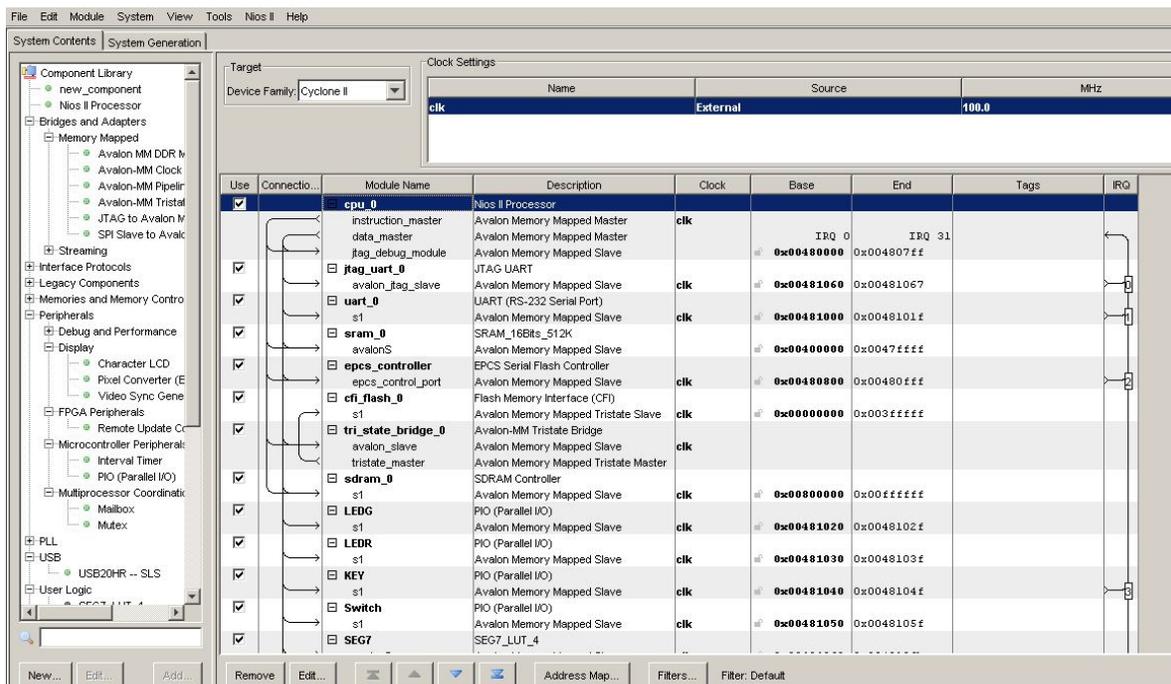


Figura 12. Configuración SOPC del sistema citado.

⁸ System on a Programmable Chip Builder

La configuración propuesta contiene como elementos importantes los siguientes:

A. Procesador NIOS II.

Se escoge una configuración para este que ofrezca las mejores prestaciones posibles, para el caso el tomado es "NIOSII/f", este procesador precisa más recursos y elementos internos de la FPGA que los demás, sin embargo la DE1 cuenta con la capacidad suficiente para soportar este procesador bajo la aplicación esperada. En la figura 13 se presenta la configuración del procesador NIOS II/f.

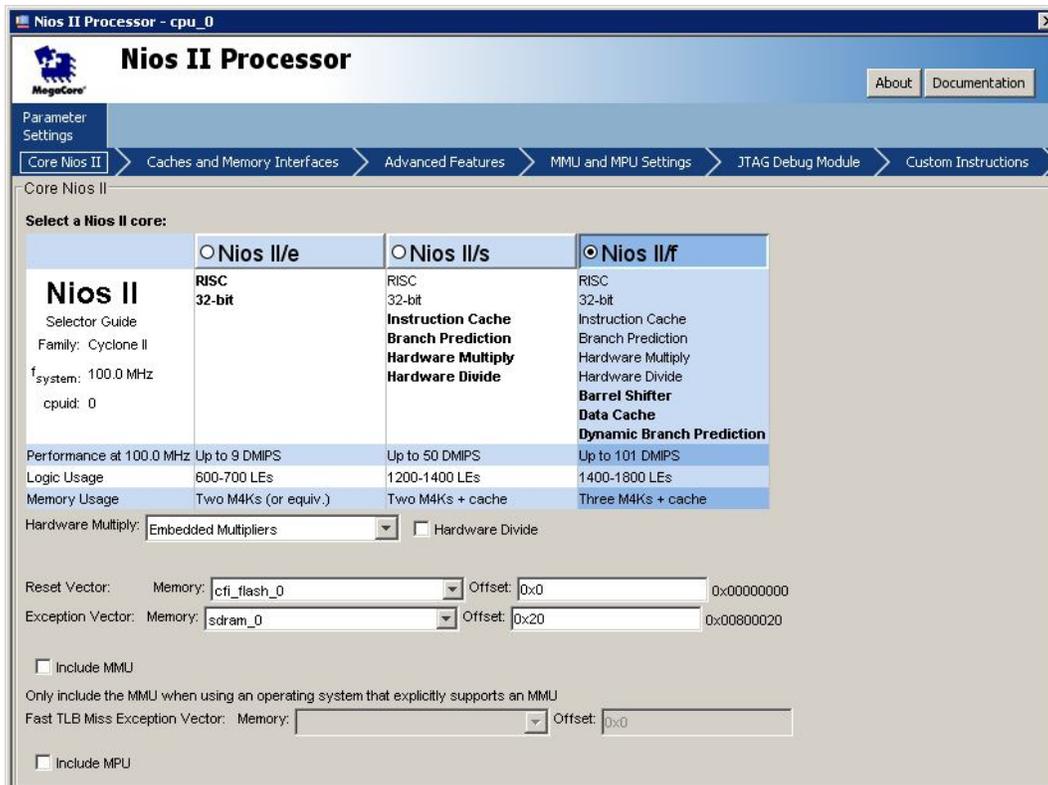


Figura 13. Configuración de Procesador NIOS II.

B. Memorias SDRAM y SRAM.

Se declaran y configuran las memorias a usar en el aplicativo, se insertan dos módulos SDRAM_0 (SDRAM controller) y SRAM_16Bits_512K, en la configuración de SOPC. En la figura 14 es posible ver la configuración de SDRAM CONTROLLER.

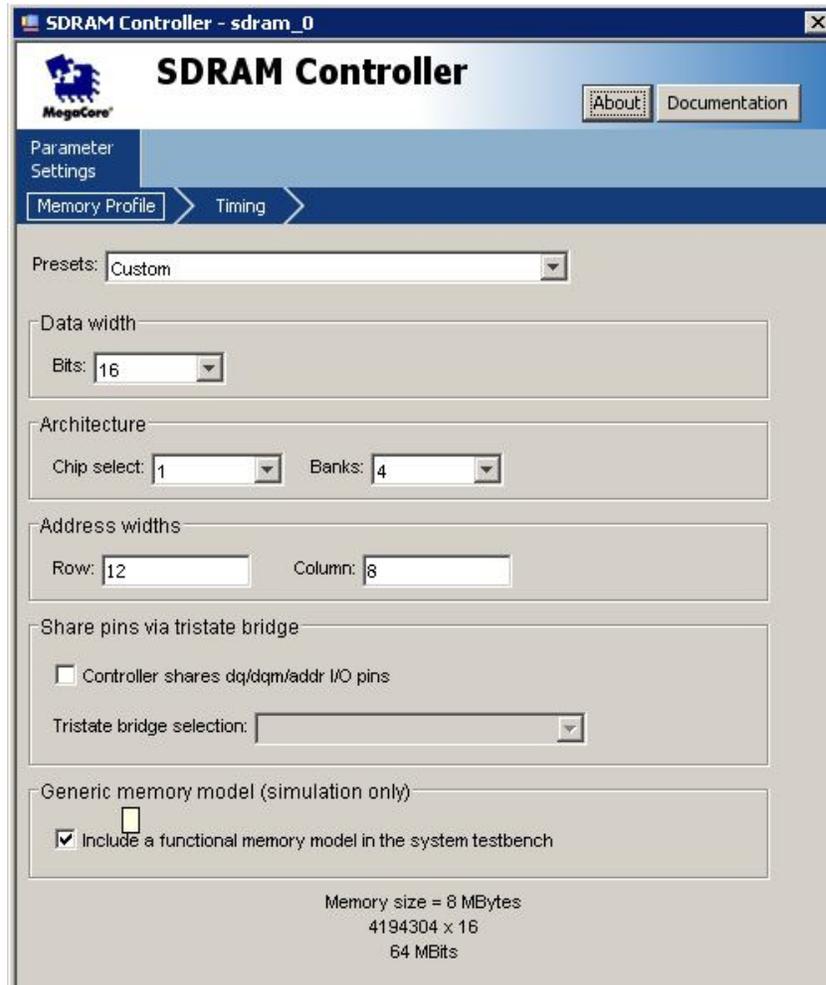


Figura 14. Configuración de SDRAM Controller.

C. Módulo UART RS232C.

Se inserta y establece un modulo para el puerto RS232 dispuesto en la DE1, para la comunicación con lo radios (salida del protocolo Modbus RTU).

Cabe anotar que esta configuración debe ser la misma para todos los esclavos, radios y maestro de la red, la configuración escogida se muestra en la figura 15.



Figura 15. Configuración de la UART, RS232C

D. Puertos de entrada y salida.

Se configuran varios puertos de salida y entrada, los cuales son los vectores de I/O, estos son los LEDR (8 led de color rojo DE1), LEDG (10 Led de color verde), Switch (10 toggles switches), Key (4 push button switches) y SEG_7 (7 segment display). En la figura 16 es posible ver la configuración de LEDR.

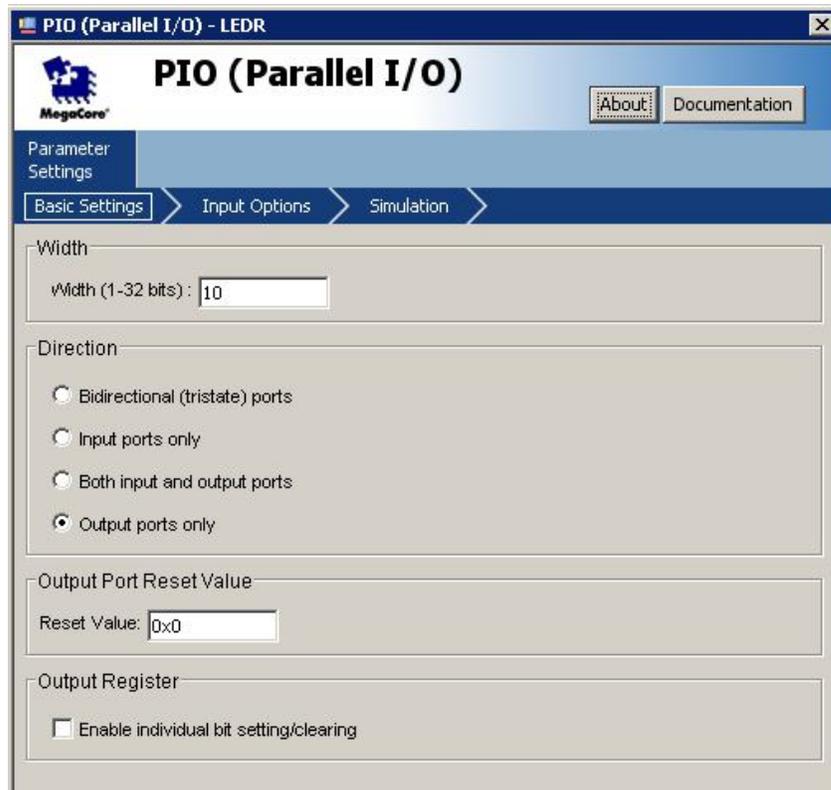


Figura 16. Configuración PIO de LEDs verdes.

E. Temporizadores.

Estos módulos son adiciones especiales de revisión de los tiempos de tramas, ya sea para conocer el fin de una y establecer comportamientos de falla por excesos de tiempos en modo de recepción de la trama. Sin estos dos bloques no es posible establecer controles de tiempos mínimos o máximos permisibles por las tramas Modbus RTU. En la figura 17 es posible ver la parametrización de uno de los temporizadores (configuración igual para los temporizados).



Figura 17. Configuración de módulos de tiempo, control de tramas.

5.1.2.1. Aplicativo de esclavo Modbus RTU en Nios II, Programación ANSI C.

El microprocesador creado sobre la tarjeta se encarga de llevar a cabo la adquisición de datos de entrada, controlar las salidas y cumplir con los requerimientos de la comunicación del protocolo (Calculo de CRC, detección de errores). Por lo cual se deben establecer secuencias, dentro de un ciclo de ejecución continuo, que realicen esta tarea en un código o programa en C. Ver anexo 1.

Se reserva una memoria para registros de recepción de datos, transmisión de datos y registros de puertos de entrada/salida.

En la figura 18, es posible ver el diagrama de flujo simplificado de acciones y decisiones del algoritmo a implementar.

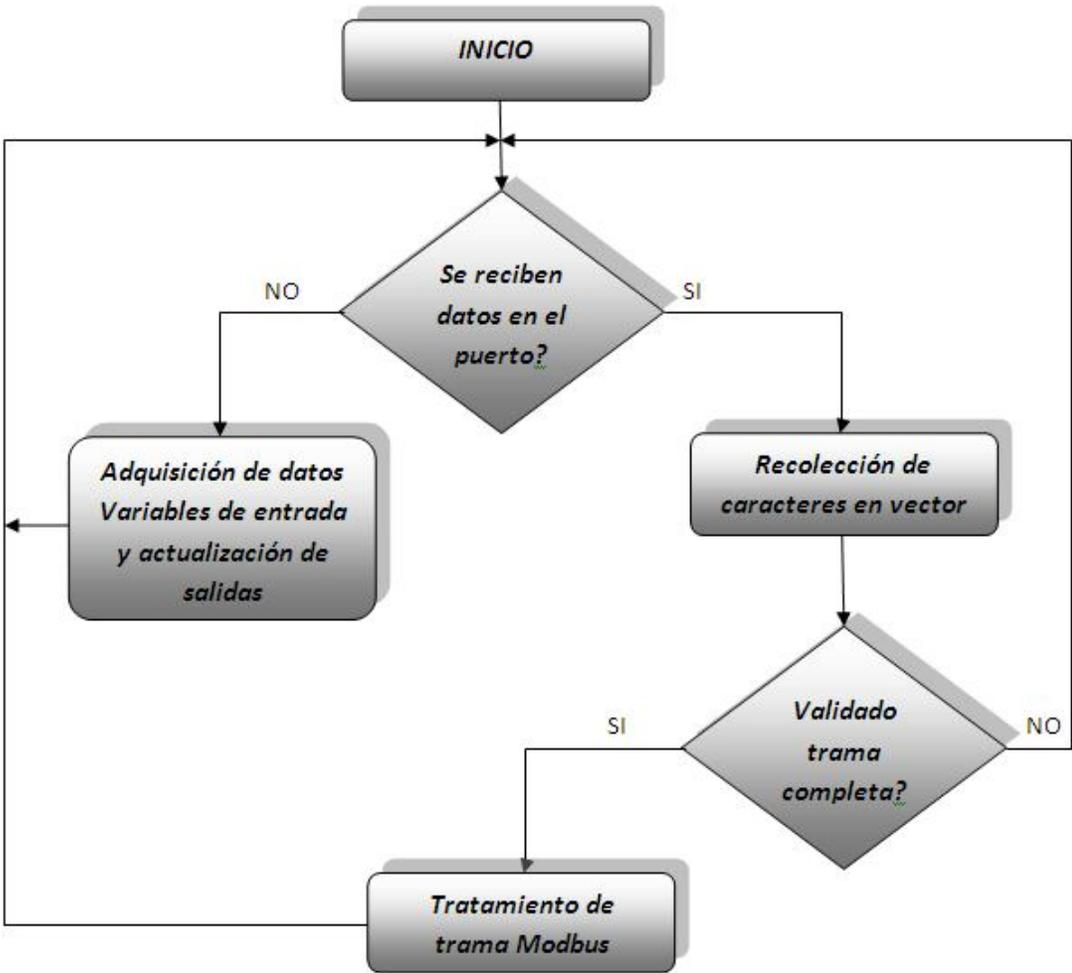


Figura 18. Algoritmo simplificado de implementado en FPGA para esclavo Modbus RTU

5.1.2.1.1. Recepción de datos por la UART RS232.

Para la recepción y validación de la consulta Modbus se implementa una función de lectura de Buffer en el programa, que recibe byte a byte la trama. Si hay un byte en el buffer de entrada de la "UART" se genera una interrupción en el ciclo de ejecución

continuo y se guarda en el buffer temporal llamado "BUFF". Si se detecta un nuevo byte en la entrada se sobrescribe en el buffer temporal "BUFF" y así sucesivamente. Cada vez que se detecta un byte se carga y se guarda hasta que se detecte otro byte que actualiza el valor del buffer temporal.

Para crear el vector de consulta MODBUS recibida se utiliza un contador de eventos llamado "COUNT" para el número de bytes detectados, por ejemplo cuando el primer byte detectado se hace "COUNT" = 1 y se guarda el byte recibido en la posición "MenRx[0]" del vector de consulta MODBUS; si este segundo byte corresponde a la dirección de la tarjeta, (validado por un corto algoritmo) entonces se procede a recibir el próximo se guarda el siguiente byte detectado, y se procede a validar el próximo como "COUNT" = 2, en la posición "MenRx[1]" y así sucesivamente, hasta que se ocupe todo el vector cuando la trama es consistente y valida.

La posición "MenRx[1]" se guarda la función Modbus solicitada la cual será la que impulse al algoritmo a una subrutina en especial. En las posiciones MenRx[2] hasta MenRx[7], se encuentran los valores de inicio, longitud y chequeo.

Cuando el valor de "COUNT" es igual a 8 indica que se ha llenado el vector de consulta "MenRx" y la consulta está lista para ser procesada por la función correspondiente.

Con el vector de Consulta MODBUS, listo para procesar, se verifican los bytes de CRC-16⁹ de entrada, si es correcto se procesa la subrutina en función de la respuesta MODBUS, en caso de no corresponder el CRC se envía un mensaje de error "CRC error" y se espera una

⁹ Cyclical Redundancy Checking

nueva trama para ser procesada. Una vez corrida la subrutinas para la respuesta a la solicitud MODBUS se hace el COUNT = 0, para que los siguientes bytes detectados se guarden en el vector de consulta MODBUS MenRx y se genere una nueva consulta. En las figuras 19 y 20 es posible ver el diagrama de flujo para la recepción, evaluación y envío de datos del protocolo Modbus RTU.

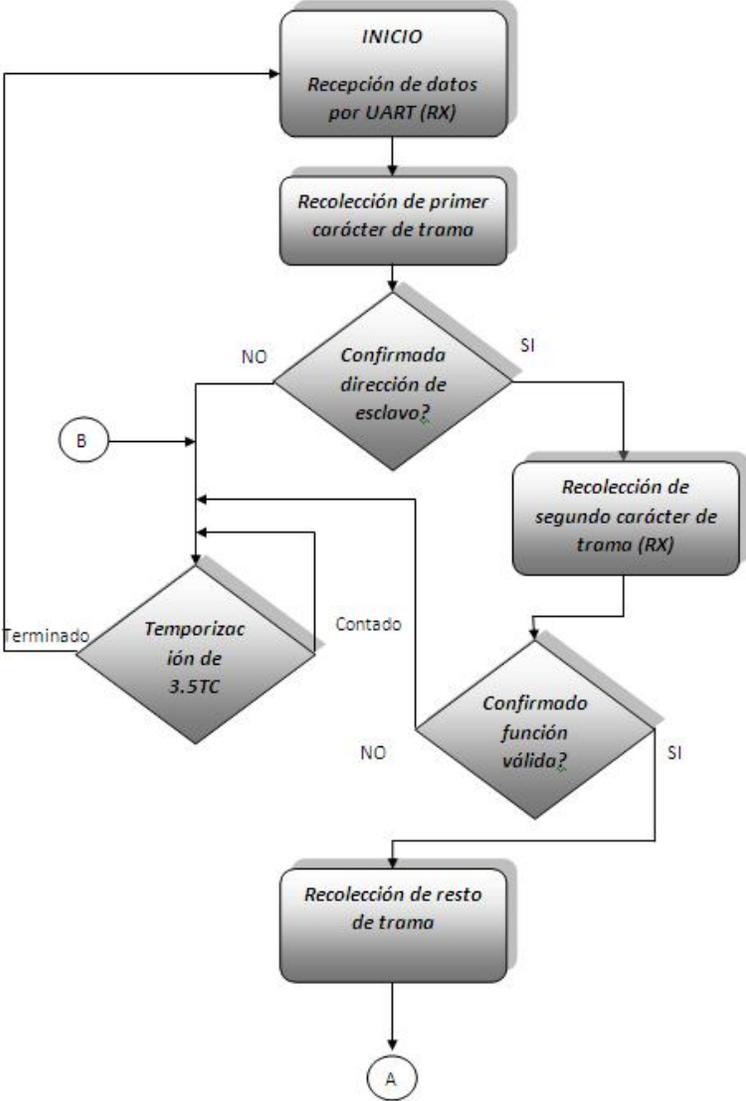


Figura 19. Diagrama de flujo para la recepción, evaluación y envío de datos Modbus RTU (1)

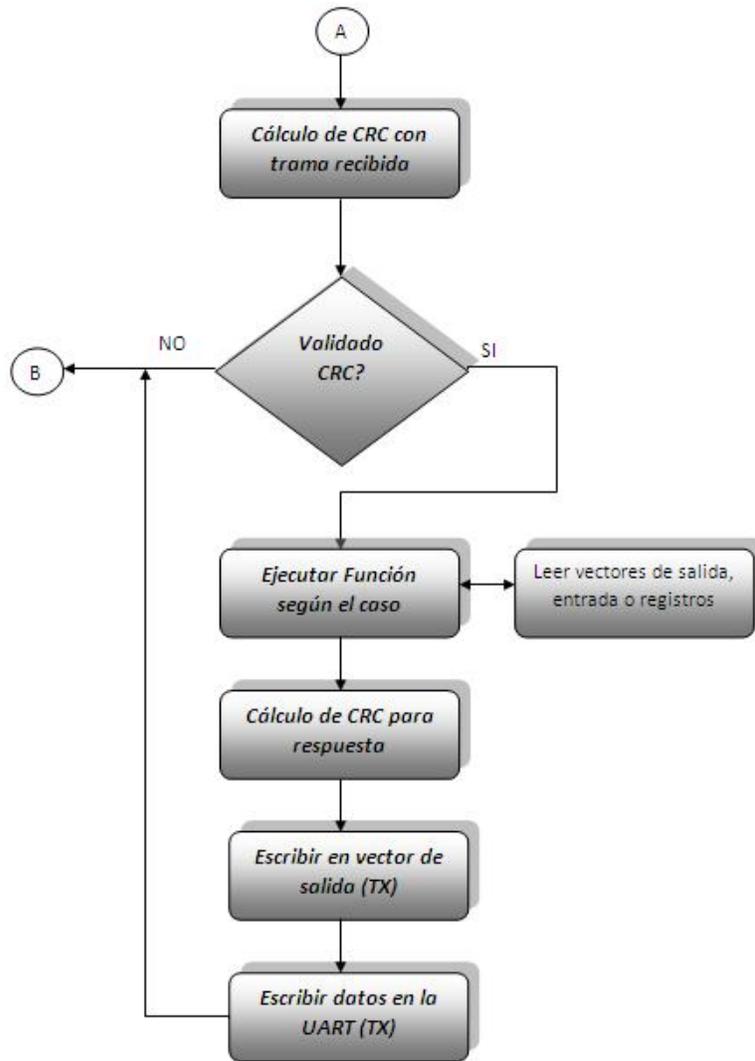


Figura 20. Diagrama de flujo para la recepción, evaluación y envío de datos Modbus RTU (2)

5.1.2.1.2. Sistema de chequeo de tramas CRC-16.

En modo Modbus RTU se incluye una instrucción de comprobación de errores que se basa en la comprobación de redundancia cíclica (CRC) método realizado en el contenido del mensaje. El campo CRC comprueba el contenido del mensaje completo. Se aplica con independencia de cualquier método de comprobación de paridad utilizada para los

caracteres individuales del mensaje. El campo CRC contiene un valor de 16 bits en ejecución como dos bytes de 8 bits [4].

Los dos bytes de CRC se adjuntan al mensaje como el último campo en el mensaje. Cuando se hace esto, el byte de orden inferior del campo es añadido en primer lugar, seguido por el byte de orden superior. El byte de CRC de orden superior es el último byte que se enviará en el mensaje [4].

Se prepara entonces una función que ejecuta la rutina para calcular el CRC, basado en los mensajes recibidos y por enviar, para el caso de mensajes recibidos el valor calculado es usado para verificar si la trama ha sido recibida sin errores, en caso de no haber validado la trama se aborta la secuencia de ejecución de funciones y se espera una nueva trama desde el maestro, en caso de ser válida se procede a elaborar las funciones solicitadas. También el CRC es calculado para el envío de mensajes, entonces el CRC hallado es colocado en las dos últimas posiciones del vector de transmisión.

5.1.2.1.3. Ejecución de las funciones de Modbus RTU.

Una vez reconocido el valor de byte "MenRx[2]", se ejecutan diferentes rutinas dependiendo de la función solicitada:

- Función 1: Lectura de Salida.

Función 1 es la lectura de estados on-off de salidas digitales [4]. El protocolo permite leer hasta 2000 estados. El registro LEDG_BASE de los LEDs verdes está configurado como único vector de salida, 8 bits o 1 byte.

Ejemplo de trama de consulta MODBUS para leer salidas (1 al 8) del único vector de salida de 8 bits del registro LEDG_BASE.

Dirección de Esclavo	Función	Dirección de Inicio HI	Dirección de Inicio LO	No. de Bobinas HI	No. de Bobinas LO	CRC16 LO	CRC16 HI
0x01	0x01	0x00	0x00	0x00	0x08	0x3D	0xCC

Ejemplo de trama de respuesta MODBUS, para la consulta.

Dirección de Esclavo	Función	Cuenta de Bytes	Bobinas 1 – 8	CRC16 LO	CRC16 HI
0x01	0x01	0x01	0x02	0xD0	0x49

- Función 2: Lectura de Entradas.

Función 2 es la lectura de estados on-off de entradas digitales [4]. El protocolo permite leer hasta 2000 estados. El registro SWITCH_BASE procede de 7 de los 10 Switch instalados tipo toggle y se ha configurado como vector de entrada de 8 bits o 1 byte. Los Switch SW0 a SW7 conforman el vector de entrada digitales usado para la función 02 de MODBUS.

Ejemplo de trama de consulta MODBUS para leer entradas digitales (1 al 8) del único vector de salida de 8 bits del registro SW_BASE.

Dirección de Esclavo	Función	Dirección de Inicio HI	Dirección de Inicio LO	No. de Bobinas HI	No. de Bobinas LO	CRC16 LO	CRC16 HI
0x01	0x02	0x00	0x00	0x00	0x08	0x79	0xCC

Ejemplo de trama de respuesta MODBUS, para la consulta.

Dirección de Esclavo	Función	Cuenta de Bytes	Entradas 1 – 8	CRC16 LO	CRC16 HI
0x01	0x02	0x01	0x5B	0xE0	0x73

- Función 3: Lectura de Registros de Salida o Registros Internos.

La función 3 hace la lectura de registros de salida o internos [4], en nuestra implementación solo se hace lectura de valores internos y calculados por la aplicación, se refieren a valores de diagnóstico de la comunicación, tales como: Porcentaje de éxito en comunicación y número de tramas corruptas. La actual aplicación solo contempla dos direcciones de 16Bits cada uno.

Ejemplo de trama de consulta Modbus-RTU con función 03, para leer los registros de diagnostico de la aplicación.

Dirección de Esclavo	Función	Dirección de Inicio HI	Dirección de Inicio LO	No. de Bobinas HI	No. de Bobinas LO	CRC16 LO	CRC16 HI
0x01	0x03	0x00	0x00	0x00	0x02	0xC4	0x0B

Ejemplo de trama de respuesta Modbus-RTU, para la consulta tipo.

Dirección de Esclavo	Función	Cuenta de Bytes	Data Hi	Data Lo	Data Hi	Data Lo	CRC16 LO	CRC16 HI
0x01	0x03	0x04	0x00	06	00	05	0xDA	0x31

- Función 4: Lectura de Registros de Entrada.

Lectura de Registros o palabras de 16 bits de entrada[4]. Con la Función 4 es posible leer los registros de un conversor análogo-digital, para leer entradas de señales análogas, los cuales son arreglables en conjuntos de 2 bytes.

Esta función direcciona a los puertos GPIO nativos dispuestos en la tarjeta.

Ejemplo de trama de consulta Modbus-RTU con función 4, para leer los registros de entrada de un conversor análogo-digital.

Dirección de Esclavo	Función	Dirección de Inicio HI	Dirección de Inicio LO	No. de Bobinas HI	No. de Bobinas LO	CRC16 LO	CRC16 HI
0x01	0x04	0x00	0x00	0x00	0x02	0x71	0xCB

Ejemplo de trama de respuesta Modbus-RTU, para la consulta tipo.

Dirección de Esclavo	Función	Cuenta de Bytes	Data Hi	Data Lo	Data Hi	Data Lo	CRC16 LO	CRC16 HI
0x01	0x04	0x04	0x00	06	00	05	0xDB	0x86

- Función 5: Forzar el cambio estado on-off de una salida.

La función 5 trata sobre el forzado del estado on-off de una salida digital [4]. El protocolo solo permite cambiar el estado de una salida por consulta MODBUS. El registro LEDG_BASE de los LEDs verdes está configurado como único vector de salida de 8 bits o 1 byte. La función 05 cambia el estado del bit correspondiente, de acuerdo a como se indique en la consulta MODBUS. Y se envía como respuesta MODBUS con función 05, la misma trama de la consulta, si se pudo llevar a cabo la operación.

Ejemplo de trama de consulta MODBUS para forzar el estado ON de una salida (1 al 8) del único vector de salida de 8 bits del registro LEDG_BASE.

Dirección de Esclavo	Función	Dirección de Bobina HI	Dirección de Bobina LO	Escribir Dato HI	Escribir Dato LO	CRC16 LO	CRC16 HI
0x01	0x05	0x00	0x01	0xFF	0x00	0xDD	0xFA

Ejemplo de trama de consulta MODBUS para forzar el estado OFF de una salida (1 al 8) del único vector de salida de 8 bits del registro LEDG_BASE.

Dirección de Esclavo	Función	Dirección de Bobina HI	Dirección de Bobina LO	Escribir Dato HI	Escribir Dato LO	CRC16 LO	CRC16 HI
0x01	0x05	0x00	0x01	0x00	0x00	0x9C	0x0A

- Función 6: Escribir un solo Registro.

Escritura de un único registro por trama de consulta Modbus-RTU en el vector de datos de salida de la función [4].

Con esta función se puede modificar datos de un sistema tal como: convertidores analógico-digital para una variable de control en campo. Dicha función escribe sobre los 2 vectores de salida de GPIO nativos de la DE1.

Ejemplo de trama de consulta Modbus-RTU tipo, para escribir un registro.

Dirección de Esclavo	Función	Dirección de Inicio HI	Dirección de Inicio LO	No. de Bobinas HI	No. de Bobinas LO	CRC16 LO	CRC16 HI
0x01	0x06	0x00	0x01	0x00	0x03	0x9A	0x9B

Ejemplo de trama de respuesta Modbus-RTU, para la consulta.

Dirección de Esclavo	Función	Dirección Hi	Dirección Lo	Escribir Hi	Escribir Lo	CRC16 LO	CRC16 HI
0x01	0x06	0x00	0x01	0x00	0x03	0x9A	0x9B

Más información sobre las direcciones y/o vectores de cada función, es posible encontrarla en la sección siguiente 5.1.2.

Para más información y detalles de la trama, consultar “MODBUS over serial line specification and implementation guide V1.02” [27].

5.1.2.1.4. Temporizaciones entre caracteres y tramas.

Dos temporizadores son usados en esta aplicación: Temporizador de 1.5 T.C¹⁰. y Temporizador de 3.5 T.C. se implementan con el Driver de tiempo Time Stamp de Nios II, que emplea un registro TIMER de 32 bits y un ciclo de reloj de 100 Mhz de la tarjeta DE1 FPGA Altera.

¹⁰ Siglas que indican tiempo de caracter.

Cuando se recibe un byte, la función "alt_timestamp_start ()" del driver cuenta el tiempo con una resolución de microsegundos, hasta que se reciba el próximo byte; hace lo misma operación con el consecuente byte y así sucesivamente. Registra el tiempo en una variable de duración de byte "td temporal", que se actualiza a medida que se van recibiendo los bytes en el Buffer de entrada. Todos los bytes recibidos se les aplica el Driver de Tiempo, independientemente si se cargan o no, en el vector de trama de consulta.

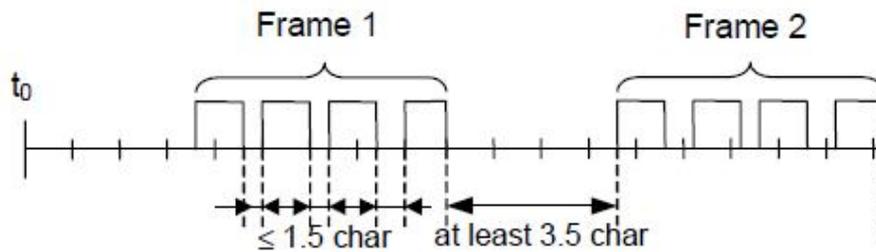


Figura 21. Tiempos entre bytes y entre tramas, soportados[4].

A. Temporizador de 3.5 T.C. (Tiempos de Carácter)

Este temporizador activa una marca cuando la duración en el tiempo entre la llegada de un carácter y el anterior recibido es mayor a 1500 μ Seg, Entonces se crea una condición de Timeout, pasando a un modo de Reinicio, listo para recibir una nueva trama de consulta. Esto puede ocurrir cuando se están recibiendo bytes en el buffer de entrada que no obedecen a una trama consistente, o un envío de un carácter aislado de error. Este temporizador vigila que el vector de "MenRX" no sea ocupado con caracteres aislados no provenientes de una trama Modbus RTU

B. Temporizador de 1.5 T.C. (Tiempos de Carácter)

Este temporizador no solo vincula el tiempo de duración de un carácter sino también la distancia en tiempo entre el siguiente más cercano.

El temporizado activa una marca cuando la duración en tiempo entre el inicio de un carácter recibido anteriormente y el actual siendo recibido no sea mayor de 500 μ Seg. Entonces se crea una condición de Timeout, pasando a modo STAND-BY (Espera), hasta que se active el temporizador de 3.5 T.C. (se ejecutaría cuando termine la recepción de una trama completa). Este vigila que los caracteres recibidos tengan la duración acordada respetando velocidad de transmisión y cantidad de bits, de esta forma no procesar bits que puedan ser originados por señales de ruido y/o errores en la comunicación, como también ayuda a alertar la ausencia de un carácter en la trama en recepción [27].

5.1.2.1.5. Aplicaciones implementadas de diagnóstico.

Junto con las funciones Modbus antes mencionadas, se implementa algoritmos de diagnóstico con el fin de facilitar la revisión de tramas exitosas recibidas.

Se adicionan las siguientes herramientas de diagnóstico:

- ü Visualización del entradas y salidas mediante led dispuestos en la tarjeta FPGA, para el caso de los vectores LEDR y LEDG.
- ü Visualización en display de la dirección configurada en el esclavo.

- ü Muestra momentánea de la función recibida con éxito en la tarjeta, esta se prepara para ser visualizada en display de 7 segmentos una vez es validada.
- ü Variable de porcentaje de éxito en comunicación con la tarjeta FPGA, monitoriable por funciones de Modbus, trata el porcentaje de tramas recibidas y acusadas con respuesta correcta. Basados en el número total de tramas recibidas. Se define por la ecuación 1:

$$\% \text{ Exitos} = \left(\frac{COUNT_FULL_FRAME \text{ (Tramas recibidas con éxito)}}{COUNT_NEW_FRAME \text{ (Tramas totales recibidas)}} \right) * 100 \quad (1)$$

- ü Con el uso de la primera tecla de “push” se pueden conocer el acumulado de tramas recibidas con error, dado la siguiente ecuación 2:

$$CORRUP_FRAMES = COUNT_NEW_FRAME \text{ (tramas totales recibidas)} - \\ COUNT_FULL_FRAME \text{ (tramas recibidas con éxito)} \quad (2)$$

Con estas funciones se facilita el conocer el estado de la comunicación en todo momento, en las pruebas y ensayos estas funciones son usadas. En la página 22 se presentan las indicaciones de diagnostico disponibles en la DE1.

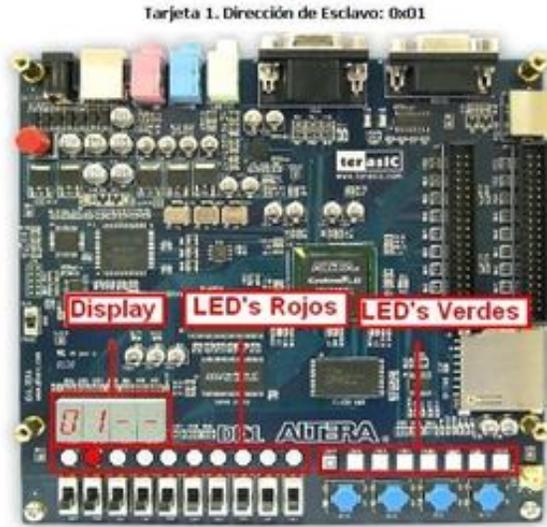


Figura 22. Indicaciones de diagnóstico disponibles en la DE1.

5.1.2.1.6. Envío de respuesta a solicitud.

De forma similar como se acumuló un vector para la recepción de bytes recibidos, en este se preparan un vector de salida para la respuesta llamado "MenTX", donde el primer byte "MenTX[0]" corresponde al número de esclavo, el segundo "MenTX[1]" a la función solicitada previamente.

Para los bytes siguientes, tanto su información como la cantidad de estos, dependen las rutinas ejecutadas en relación directa con la función ejecutada. Cada función es la responsable de llenar el vector de transmisión, a partir del MenTX[2]. Los últimos dos bytes de la trama corresponden al CRC-16 [4].

Una vez con este vector dotado de la información de envío, se abre el puerto UART para la salida de los datos hacia el Maestro.

5.1.3. Mapa de memorias de la implementación realizada.

Después de la implementación del sistema, la tarjeta está lista para ser interrogada bajo el uso del protocolo, por lo tanto se hace necesario conocer los mapas de memoria para cada una de las funciones disponibles, en la tabla 9 se resume el mapa de memorias físicas y lógicas de la FPGA.

Mapa General de Memorias							
Tipo de Registro FPGA	Función permitida	Direcciones de Registro	Tipo	Dirección física en FPGA	Formato	Valores en Decimal	Observación
Salidas digitales	01 (lectura Bit)	10000 – 10007	Salidas	Salidas LedR0- LedR7	1 bit	0, 1	lectura de salidas nativas de la FPGA en sus led Verdes, ver mapa función 01
Entradas digitales 1	02 (lectura de bit)	20000 – 20009	Entrada	Entrada SW0- SW7	1 bit	0, 1	lectura de entradas a swichts nativos de la FPGA
Salidas Digitales	03 (lectura registros internos)	30000 – 30050	Salidas	Memoria	16 bits	0 a 65535	lecturas de registros. Ver mapa función 03
Entrada analoga	04 (lectura registros Entrada)	40000 – 40050	Entrada	Memoria	16 bits	-10000 a 10000	disponibles solo en memoria, no cableados
Salidas Digitales	05 (escritura Bit)	50000 – 50007	Salidas	Salidas LedR0- LedR7	1 bit	0, 1	escritura de salidas nativas de la FPGA en sus led Verdes
Salidas analogas	06 (Escribir registro salida)	60000 – 60015	Salidas	Memoria	16 bits	-10000 a 10000	Escritura de salidas nativas de la FPGA

Tabla 9. Mapa General de Memoria

5.1.3.1. Mapa de memoria para la función 01.

A continuación, en la tabla 10 es posible apreciar el mapa de memoria de la función 01 implementada, la cual es la lectura de salidas digitales, sobre el vector de led rojos dispuestos en la tarjeta DE1.

Mapa de memoria de FPGA para la función 01			
Dirección Modbus	Descripción	Valores	Observación
10000	Led Rojo 0	0, 1	Estado del led rojo nativo 0
10001	Led Rojo 1	0, 1	Estado del led rojo nativo 1
10002	Led Rojo 2	0, 1	Estado del led rojo nativo 2
10003	Led Rojo 3	0, 1	Estado del led rojo nativo 3
10004	Led Rojo 4	0, 1	Estado del led rojo nativo 4
10005	Led Rojo 5	0, 1	Estado del led rojo nativo 5
10006	Led Rojo 6	0, 1	Estado del led rojo nativo 6
10007	Led Rojo 7	0, 1	Estado del led rojo nativo 7

Tabla 10. Mapa de memoria para la función 01.

5.1.3.2. Mapa de memoria de la función 02.

La función 2 corresponde a lectura de entradas, en el vector de Led verdes, llamado LedG.

La FPGA dispone de 9 pulsadores, los cuales en esta implementación son las entradas provenientes del proceso, a continuación la tabla 11, especificando su dirección para lectura.

Mapa de memoria de FPGA para la función 02				
Dirección Modbus	Descripción	Valores	Estado	Observación
20000	SW0	0, 1	ON/OFF	Estado del Switch nativo 0
20001	SW1	0, 1	ON/OFF	Estado del Switch nativo 1
20002	SW2	0, 1	ON/OFF	Estado del Switch nativo 2
20003	SW3	0, 1	ON/OFF	Estado del Switch nativo 3
20004	SW4	0, 1	ON/OFF	Estado del Switch nativo 4
20005	SW5	0, 1	ON/OFF	Estado del Switch nativo 5
20006	SW6	0, 1	ON/OFF	Estado del Switch nativo 6
20007	SW7	0, 1	ON/OFF	Estado del Switch nativo 7

Tabla 11. Mapa de memorias para la función 02.

5.1.3.3. Mapa de memoria de función 03.

La función 03 trata sobre la lectura de registros de salida o internos, para el caso de la implementación el vector a leer en la tarjeta corresponde este corresponden a variables de diagnóstico preparadas especialmente para determinar el estado de la comunicación inalámbrica en el punto de instalación, en la tabla 11 es explicado las variables internas. En la sección 5.1.1 es explicado el cálculo de cada una de las variables.

Mapa de memoria de FPGA para la función 03				
Dirección Modbus	Descripción	Valores	Unidad	Observación
30000	Porcentaje de Éxitos Comunicación	0 – 1000	%	Divisible entre 10, 1000=100.0%
30001	Número de tramas corruptas	0 – 65535		acumulado numero de tramas corruptas
30002 - 30050	Disponibles	-		No programadas

Tabla 12. Mapa de memorias para la función 03.

5.1.3.4. Mapa de Memoria de función 04.

Con la función 04 es posible leer registros de entrada en la FPGA, los cuales para la implementación han sido dispuestos los puertos GPIO-1 y 2. Desde este punto es posible leer registros procedentes de otras tarjetas conversores Análogos/digitales, la recepción de información de estos puertos es de forma paralela y un solo registro de 16 bits por puerto. La tabla 13 muestra las variables.

Mapa de memoria de FPGA para función 04.				
Dirección Modbus	Descripción	Valores	Formato	Observación
40000	GPIO1-0 -- GPIO1-63	0 - 65535	16 Bits	Estado del Puerto Nativo GPIO-1
40063	GPIO1-64 -- GPIO1-127	0 - 65535	16 Bits	Estado del Puerto Nativo GPIO-2

Tabla 13. Mapa de memorias para la función 04.

5.1.3.5. Mapa de memoria para función 05.

La función escribe en variables de salida de longitud de un 1 bit, por lo tanto esta se implementa para escribir sobre las salidas de la FPGA en el vector LEDR, la tabla 14 resume las variables de salida a la cuales es posible escribir.

Mapa de memoria de FPGA para función 05.				
Dirección Modbus	Descripción	Valores	Unidad	Observación
50000	Led Verde 0	0, 1	Bit	Forza Estado del led Verde nativo 0
50001	Led Verde 1	0, 1	Bit	Forza Estado del led Verde nativo 1
50002	Led Verde 2	0, 1	Bit	Forza Estado del led Verde nativo 2
50003	Led Verde 3	0, 1	Bit	Forza Estado del led Verde nativo 3
50004	Led Verde 4	0, 1	Bit	Forza Estado del led Verde nativo 4
50005	Led Verde 5	0, 1	Bit	Forza Estado del led Verde nativo 5
50006	Led Verde 6	0, 1	Bit	Forza Estado del led Verde nativo 6
50007	Led Verde 7	0, 1	Bit	Forza Estado del led Verde nativo 7

Tabla 14. Mapa de memoria para la función 05.

5.1.3.6. Mapa de memoria para función 06.

La función se implementa para la escritura de registros de salida, por lo cual es útil para la forzado de variables de salida, en especial sobre el GPIO -1 y 2, de la tarjeta FPGA . Estos puertos pueden ser destinados para conexión con conversores digitales análogos, por conexión paralela. En la tabla 15 se describen las variables.

Mapa de memoria de FPGA para función 06.				
Dirección Modbus	Descripción	Valores	Formato	Observación
40000	GPIO1-0 -- GPIO1-63	0 - 65535	16 Bits	Forza Estado del Puerto Nativo GPIO-1
40063	GPIO1-64 -- GPIO1-127	0 - 65535	16 Bits	Forza Estado del Puerto Nativo GPIO-2

Tabla 15. Mapa de memoria para la función 06.

5.2. Configuración de Radios FHSS y enlace radio.

Los radios XStream-PKG ofrecen la técnica de modulación FHSS explicada en capítulos anteriores. Una configuración óptima de este equipo en la implementación es la de fábrica, por lo cual se puede establecer desde el primer momento de su encendido el enlace entre maestro y esclavos, dicha comunicación se basa en la facilidad como se puede enviar un carácter y este es replicado en los pines de salida de los demás radios instalados en la cercanía. Es decir, todas las estaciones esclavas escucharán los caracteres enviados por alguna estación que envíe en ese momento una trama. El protocolo facilita que los mensajes solo sean procesados por la estación correspondiente, dado que el primer byte de la trama expresa el número del esclavo, por lo tanto el esclavo que no tenga configurada dicha dirección no procesará esta información. Cabe anotar que las direcciones de esclavos son inequívocas.

Adicionalmente la transmisión de datos se hace mediante "broadcast"¹¹, por lo tanto todo radio dispuesto en el radio de alcance recibirá y colocará la información recibida en su puerto RS232 para la estación o tarjeta DE1, no existe reconocimiento por parte del radio de comunicación, este es solo es medio de transmisión similar al uso de cable entre las tarjetas DE1. Dado que no existe reconocimiento ni tampoco retransmisión controlada de forma independiente por el radio, la responsabilidad de controlar la ejecución y re-envío de tramas recae sobre el maestro, que conoce desde el punto de aplicación, el tipo de información que desea conocer y de qué esclavo en especial.

¹¹ Trasmisión de datos a todos los receptores cercanos, sin reconocimiento.

5.2.1. Configuración de Radio FHSS.

La configuración hardware que permite las comunicaciones radio antes mencionada es la siguiente:

- a. Configuración de DIP-Suiches, figura 23.

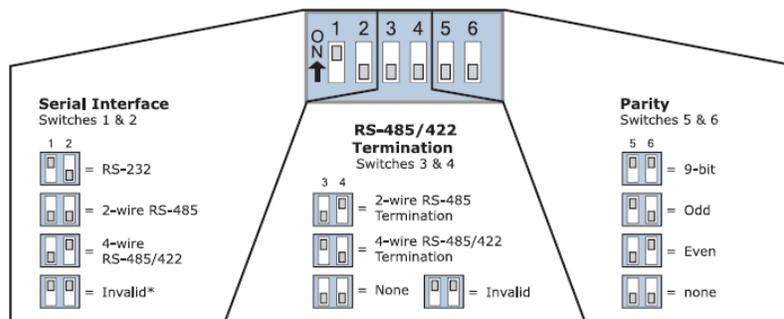


Figura 23. Suiches de configuración de radios.[2]

Por conveniencia de los puertos dispuestos se preparan los radios con:

- SW 1, 2: Interfase RS232. Para el caso del esclavo se debe usar un conversor “Null Modem”
- SW 3, 4: Resistencia terminal, None.
- SW 5,6: Paridad, None.

Adicionalmente se hace necesario preparar el cableado de conexión a RS232 entre la DE1 y los radios de FHSS, el cual es ilustrado a continuación, figura 24.

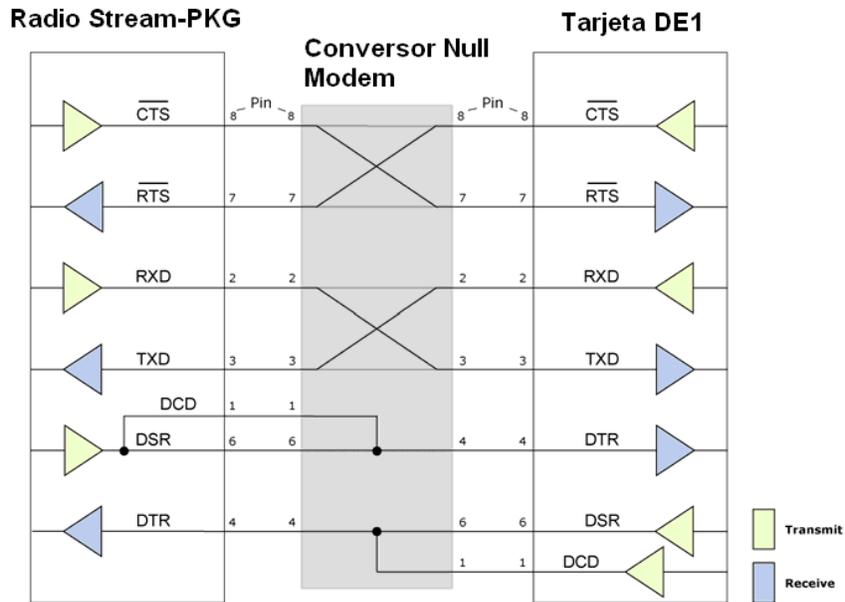


Figura 24. Configuración de Cableado de Radio a DE1 (FPGA)[2]

5.2.2. Enlace de comunicación radio.

El enlace radio se hace 900Mhz usando la banda ISM (902-928MHZ), los radios transmiten lo recibido en su puerto RS232 a 19.2Kb, con la configuración de paridad antes presentada, ver figura 25.

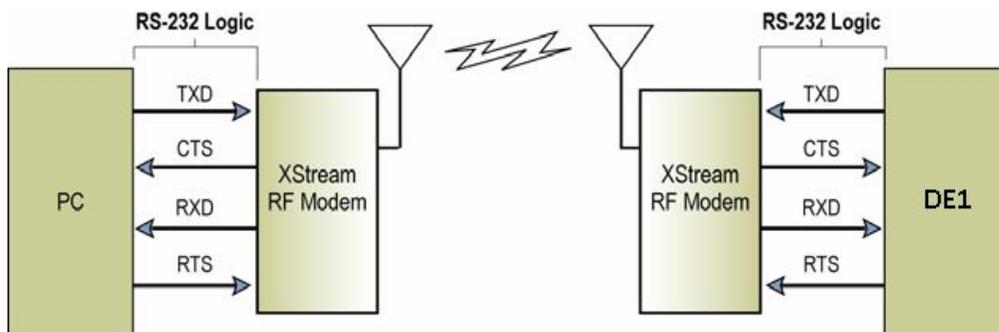


Figura 25. Esquemático de enlace radio.[2]

Cada radio tiene 4 estados de operación, los cuales se controlan de forma automática, se presentan en la figura 26.

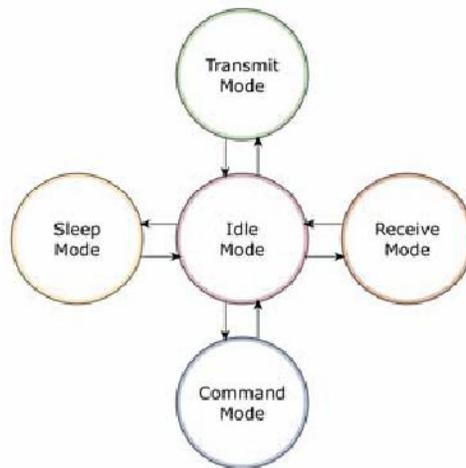


Figura 26. Estados de operación, radios FHSS[2].

Los estados de transmisión, recepción y libre (Idle), son tratados más adelante, los estado de comando e inactividad (sleep) no son necesarios ni desarrollados para esta implementación.

5.2.2.1. Estado Libre (Idle):

Es un estado de reposo en el que el equipo no se encuentra recibiendo ni enviando información, el equipo hace automáticamente la transición a otro estado dependiendo de la condición que se presente, tales condiciones son [2]:

- Dato recibido en el puerto RS232, logra el cambio a modo de transmisión.
- Dato recibido en antena, hace el cambio a modo de recepción.

Una vez terminado estos estados automáticamente cambia a modo de reposo.

5.2.2.2. Estado de transmisión:

En este estado son enviados los caracteres recibidos desde el puerto RS232.

Una vez validado que es recibido un carácter este prepara una secuencia de sincronización con los radios receptores, con el fin de reservar los canales de comunicación, mientras este proceso se lleva a cabo, el radio puede seguir recibiendo información del puerto.

La longitud máxima del paquete es de 64bytes, y consta de un inicializador que lleva la información de saltos de frecuencia y variables de acople junto con los datos útiles que en realidad se desea enviar (se constituye de encabezado, datos recibidos en el puerto RS232 y CRC) en caso de ser necesario enviar más información, dado que no se ha vaciado el buffer de recepción, se realiza tiempo después un nuevo envío [2]. Ver figura 27.

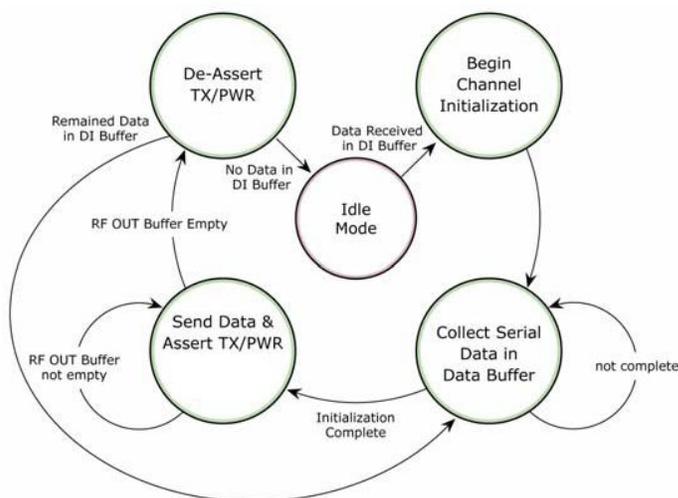


Figura 27 .Proceso de transmisión mensajes [2].

5.2.2.3. Estado Recepción:

Si el radio se encuentra en modo de reposo y recibe la un byte por el puerto de la antena, inmediatamente pasa a modo de recepción. Una vez recibida la totalidad de la trama procede a validar lo recibido revisando el CRC, si este es correcto entrega el paquete útil de datos al buffer del puerto RS232, en caso de no serlo, descarta todo el mensaje.

En este estado el radio consume la misma energía que el estado de reposo [2].

6. Pruebas y ensayos del sistema dentro de un ambiente industrial.

Con el fin de establecer de mejor manera las ventajas del sistema implementado, realizaremos un conjunto de pruebas, basados en los siguientes requisitos previos:

- Configuración de un PC maestro para la lectura de las variables dispuestas en la FPGA por medio de las funciones básicas de Modbus RTU, variables que especial que este sistema vigilará son las estadísticas de tramas recibidas exitosamente.
- Vinculación del sistema a un proceso de producción, el cual es el monitoreo de estado de niveles de los tanques (mediante señales on –off), para la indicación de presencia o ausencia de materiales. Proceso con el cual se busca que ofrezca cambio en los estados de las señales a ser recibidos por la FPGA. Cabe anotar que cualquier proceso productivo es útil para dicho ensayo, siempre y cuando las dificultades en las condiciones cableado y/o accesos de dichas señales sean en realidad el parámetro que resalte para ser de la transmisión inalámbrica una solución de monitoreo.
- Reconocimiento de zonas, las cuales se destacan por ser generadoras de ruido eléctrico como también aquellas que son susceptibles a perturbaciones por esta clase ruidos.

6.1. Generalidades de la Prueba.

Es muy importante establecer de manera general los puntos en donde se ejecutan las pruebas de comunicación, con el fin de enmarcar las mismas dentro del entorno de estudio necesario.

La planta usada para las pruebas es BIOFILM S.A., productora de película de Polipropileno bioorientado ubicada en la ciudad de Cartagena, la compañía cuenta con tres líneas de producción automatizadas y controladas desde cuartos de control donde están dispuestos los correspondientes controladores y redes de comunicación, en su mayoría Profibus DP¹², para el caso de la prueba se usará la línea de mayor potencia y mas número de electrónica asociada.

A lo largo del recorrido de la señal de comunicación a ensayar se encuentran equipos generadores de ruido eléctrico (algunos descritos en la tabla 4) como también equipos que de cierta forma son sensibles a interferencias, tales como módulos electrónicos, dispositivo de control automático, entre otros. En las figuras 28, 29, 20 y 31 se presentan algunas de los equipos de la planta.

¹² Profibus DP: Tipo de red de comunicación industrial muy adoptado en el nivel de campo, principal proveedor Siemens.



Motores Eléctricos con
variación de velocidad,
bajo control PWM

Figura 28. Motor AC de 400Kw con variación de velocidad por frecuencia.



Variador de velocidad
para motores AC.

Figura 29. Variador de velocidad AC.



Serie de motores trifásicos jaula de ardilla, de arranque directo

Figura 30. Set de motores de ventilación para horno.



Modulo ET-200M, fabricante Siemens, para la recolección de señales de campo provisto con Profibus DP

Figura 31. Módulos de campo con comunicación industrial.

En las pruebas a realizar se espera no ser afectado y ni afectar equipos dispuestos en la planta, sin embargo no se tratará de aislarlos ante los efectos de dicha prueba. En la figura 32 se describen sus ubicaciones dentro de un sector de la planta, y en la tabla 16 se resumen los tipos de fuentes.

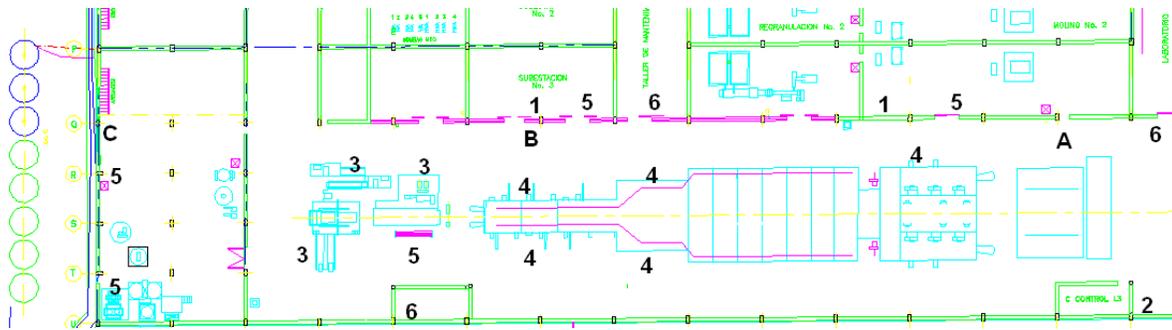


Figura 32. Ubicación de equipos en planta, prueba de comunicación

DESIGNACION	SIGNIFICADO
1	Presencia de variadores de velocidad AC, con retroalimentación de encoder y comunicación Profibus DP
2	Controladores lógicos Programables PLC, con comunicación Profibus DP y Ethernet
3	Motores eléctricos AC, de inducción, con variación de velocidad, potencias entre los 30Kw a 400Kw
4	Motores eléctricos AC, de inducción, sin variación de velocidad, de potencias entre los 7Kw y 15kw.
5	Estaciones electrónicas de recolección de señales de campo, con comunicación Profibus DP
6	Fuentes de redes inalámbricas corporativas, WIFI, IEEE802.11
A	Maestro Modbus RTU para pruebas de comunicación internas de estabilidad
B	Esclavo 1 Modbus RTU para pruebas de comunicación internas de estabilidad
C	Esclavo 2 Modbus RTU para pruebas de comunicación internas de estabilidad

Tabla 16. Designación para ubicación en diagrama de planta.

Se realizan tres mediciones importantes: medición de recursos empleados en la FPGA, paquetes recibidos vs distancia y estabilidad en comunicaciones:

6.2. Medición de recursos a tarjeta DE 1.

Con el uso del Software Quartus se logra medir los recursos empleados por la aplicación Modbus RTU, en la figura 32 es posible ver los resultados.

Flow Status	Successful - Wed Oct 20 18:09:25 2010
Quartus II Version	9.0 Build 132 02/25/2009 SJ Full Version
Revision Name	DE1_NIOS
Top-level Entity Name	DE1_NIOS
Family	Cyclone II
Device	EP2C20F484C7
Timing Models	Final
Met timing requirements	Yes
Total logic elements	4,651 / 18,752 (25 %)
Total combinational functions	3,960 / 18,752 (21 %)
Dedicated logic registers	2,685 / 18,752 (14 %)
Total registers	2802
Total pins	283 / 315 (90 %)
Total virtual pins	0
Total memory bits	75,264 / 239,616 (31 %)
Embedded Multiplier 9-bit elements	4 / 52 (8 %)
Total PLLs	1 / 4 (25 %)

Figura 32. Resultado de Quartus, aplicativo Modbus RTU

El aplicativo preparado consume aproximadamente el 25% de recursos lógicos dispuestos por la DE1, y 90% de los pines de hardware como era de esperarse dada la cantidad de conexiones realizadas en pro de un aplicativo útil. Los valores son satisfactorios para poder correr la aplicación implementada.

6.3. Pruebas de comunicación a la implementación.

Se dotó de un pequeño algoritmo adicional en la FPGA para el diagnóstico en línea del estado de la comunicación, la cual buscaba arrojar como valor de salida el porcentaje de tramas recibidas sin errores, ver ecuaciones 3 y 4.

$$\% \text{ Exitos} = \left(\frac{COUNT_FULL_FRAME \text{ (Tramas recibidas con éxito)}}{COUNT_NEW_FRAME \text{ (Tramas totales recibidas)}} \right) * 100 \quad (3)$$

$$CORRUP_FRAMES = COUNT_NEW_FRAME \text{ (tramas totales recibidas)} - \\ COUNT_FULL_FRAME \text{ (tramas recibidas con éxito)} \quad (4)$$

Con dichas ecuaciones se busca realizar dos tipos de pruebas, con variación de distancia y estabilidad de comunicación.

Dado que se dispone de radios de comunicación FHSS, los cuales ofrecen buenas prestaciones pese al ruido electromagnético presentes en la industria, se procede a hacer pruebas en dos escenarios, interno de planta (con presencia de señales de interferencia de banda angosta y banda ancha) y exterior de planta (poca presencia de interferencias) y así ver sus diferencias. Las pruebas se realizaron comprobando el porcentaje de tramas recibidas de forma óptima a diferentes distancias (solo para el caso de pruebas de variación de distancia) y tramas recibidas de forma errónea (solo para el caso de prueba interna de estabilidad).

Las pruebas se ejecutaron bajo las siguientes condiciones: frecuencia de transmisión de 900Mhz, 100mw de potencia en antena y línea de vista siempre mantenida, adicionalmente temperatura y humedad propias de una planta situada a la cercanía del mar y a cero metros altura.

Para tales fines se dispone de un maestro PC, con software industrial llamado "Protool Pro" [26] de marca Siemens, configurado de tal manera que pueda ser el maestro de comunicaciones y asociable a los esclavos FPGA desarrollados, este ejecutará las consultas al esclavo bajo el conocimiento de las variables disponibles, en la figura 33 se encuentra la pantalla de visualización de variables monitoreadas por el sistema.

El valor de aciertos y tramas erróneas también están disponibles en el monitor del maestro PC dado que está disponible como variable de una de las funciones Modbus, ver figura 33.

IMPLEMENTACIÓN DEL PROTOCOLO MODBUS-RTU

UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
PROGRAMA DE MAESTRÍA EN INGENIERÍA
CARTAGENA, FEBRERO 10 DE 2010

Prueba Industrial > #8 9-02-2010

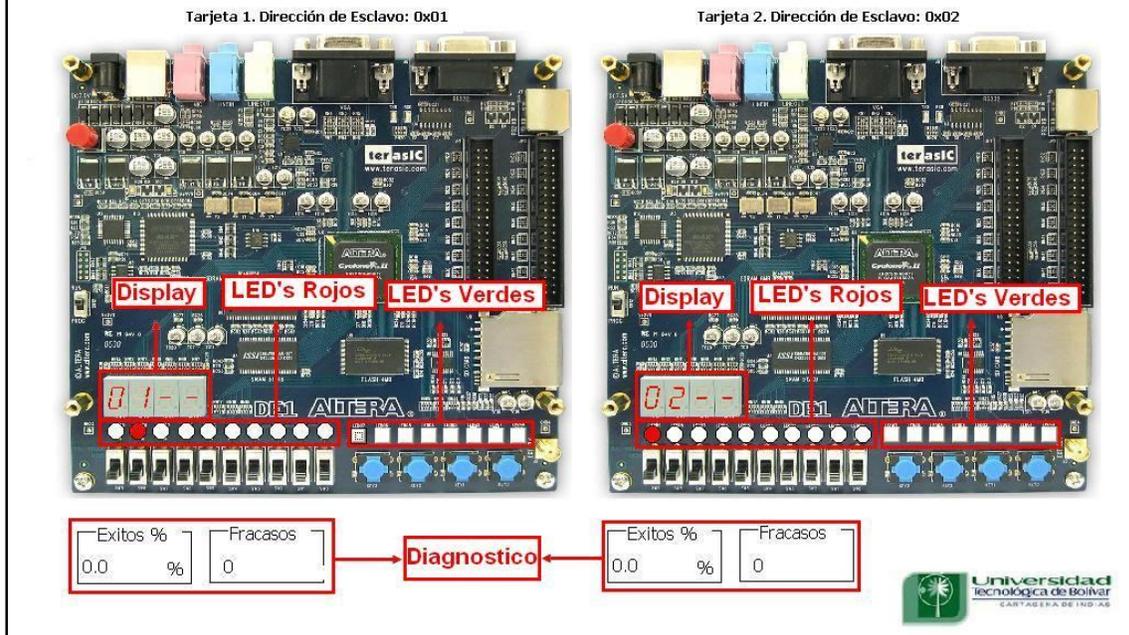


Figura 33. Muestra de pantalla de software maestro para pruebas de comunicación, configurado para el desarrollo de las pruebas en "Protool Pro" de Siemens.

Adicionalmente se realizaron medidas de temperatura y humedad también versus distancia obteniéndose las siguientes mediciones.

6.3.1. Pruebas al interior de planta (variación de distancia).

En el interior de planta se encuentran presentes muchas modalidades de ruido, descritas en capítulos anteriores como son los llamados: banda ancha y banda estrecha, dada la presencia de variados motores que llegan hasta 480Kw, sistemas de variación de velocidad, fuentes de conmutación electrónica y demás equipos de conmutación a

diferentes frecuencias. Bajo este entorno se realizan las pruebas tomando como variable independiente la distancia entre el maestro y esclavo Modbus, y como variable dependiente la cantidad de tramas recibidas con éxito.

Mediante la aplicación de diagnóstico se prepara la prueba modificando la distancia entre los sistemas, PC (maestro) y esclavo, y se procede a tomar datos con aumentos en distancia en pasos de 8 metros del esclavo a maestro, previamente enlazados por medio de los radio con modulación FHSS.

Se obtiene el siguiente comportamiento, figura 34.

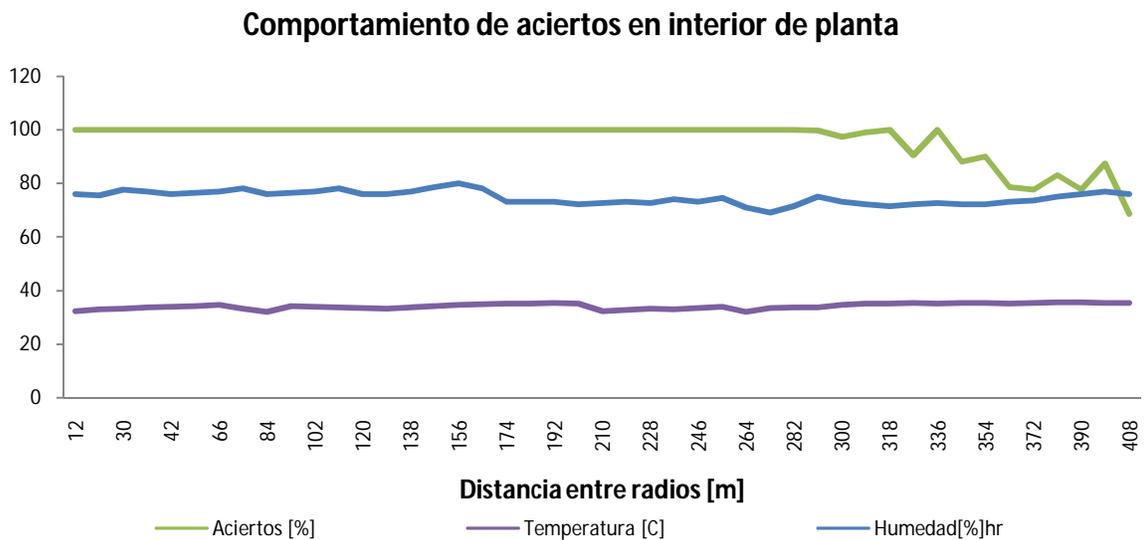


Figura 34. Comportamiento de aciertos en tramas en interior de planta.

6.3.2. Pruebas al exterior de planta (variación de distancia).

De forma similar se realizaron mediciones al exterior de la planta, alejado de los que podría ser interferencias con dispositivos de generación de ruido en banda ancha

especialmente, estableciendo línea de vista entre los radios se procedió a tomar el porcentaje de aciertos de tramas recibidas, temperatura y humedad relativa. Los resultados son presentados en la figura 35.

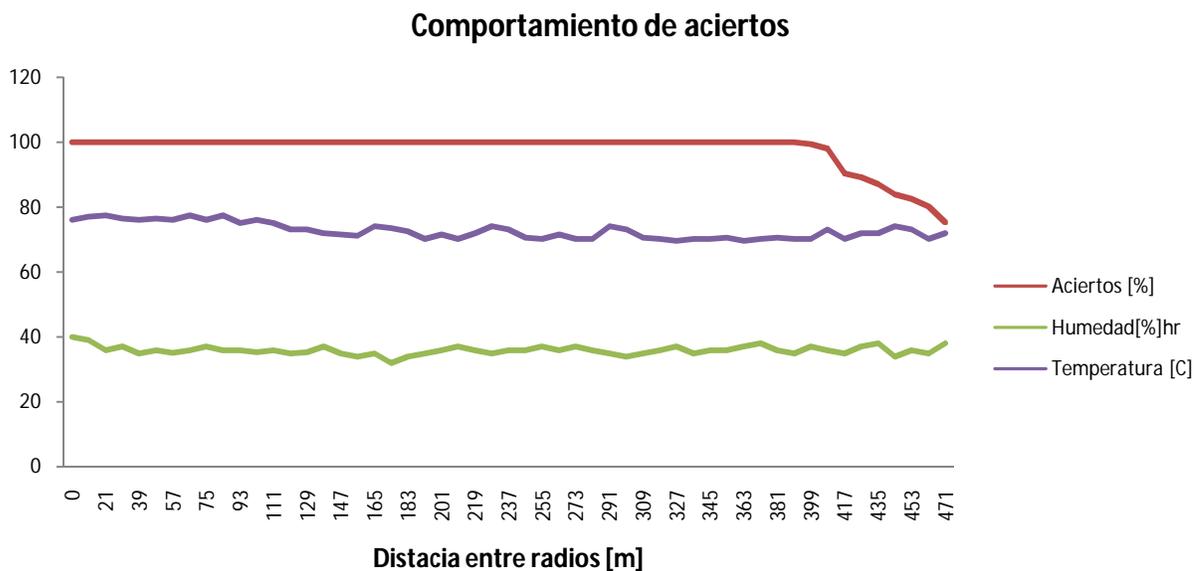


Figura 35. Comportamiento de aciertos en tramas en exterior de planta.

6.3.3. Prueba de estabilidad de comunicación.

La prueba de estabilidad consistió en mantener el sistema en comunicación, dotado de un maestro y dos esclavos por un rango de tiempo suficiente para poder tener una aproximación de su comportamiento fijando especial atención en la cantidad de errores en tramas recibidos en su operación al interior de planta (situación con la mayor cantidad de interferencias).

Se mantuvo a dos esclavos durante 8 días, tiempo continuo enlazado con su maestro (PC con software "Protool Pro", a una distancia aproximada de 100 metros para un esclavo y 200 metros para otro, con 100 metros entre estos últimos.

Bajo el uso de la ecuación (4) en la FPGA, se obtuvo un error para el esclavo más cercano y dos para el más lejano al cabo del tiempo de prueba (8 días), pero estos errores no fue suficientes para agotar el timeout dispuesto por defecto en el Maestro PC, por lo tanto dicho equipo no mostró error a los largo de la operación bajo esta prueba.

Basados en que el maestro realiza solicitudes de lectura cada 0.5 segundos para casi la totalidad de las variables (solo las variables de lectura de diagnóstico fueron configuradas en 2 segundos) de un total de 13. Por lo tanto casi 18 millones de solicitudes fueron realizadas en esos términos a cada estación, lo cual arroja como porcentaje de falla en condición estable al Esclavo 1 (200metros) de $5.5e-6\%$ de error y a la estación 2 (100metros) de $1.1e-5\%$ de error.

7. Conclusiones y trabajos futuros.

Como principal conclusión, nos debemos remitir a la pregunta de investigación planteada desde el inicio de esta investigación (presentada en el capítulo de introducción) con el fin de darle solución, su respuesta inicia con la definición de confiabilidad expuesta por Howitt. I. [28], la cual se refiere a confiabilidad como la capacidad de la red para llevar a cabo sus necesidades funcionales transportando la información de la aplicación industrial en la red dentro de un amplio rango de condiciones operativas; podemos postular que unas de las más relevantes condiciones de operación refiere al medio en el cual se desempeña la comunicación, la cual en este caso es un medio industrial con presencia de interferencias emitidas por diferentes fuentes. La comunicación gana mucha confiabilidad al obtener un grado alto de inmunidad con la presencia de estas clases de interferencias si se sostiene bajo modulación FHSS [20][22], por lo tanto la confiabilidad de protocolo Modbus RTU en un enlace inalámbrico es sostenible en un ambiente industrial bajo el uso de una estrategia de modulación FHSS.

A continuación se describen algunas de las conclusiones obtenidas una vez terminada y probada la estación experimental, por lo tanto es posible exponer:

- Modbus-RTU es un protocolo relativamente simple, que requiere pocos recursos de hardware para su implementación, por ser uno de los protocolos mas documentados se facilita mucho su ejecución en dispositivos como lo es un FPGA, en la implementación de la estación esclava Modbus RTU se llegó a utilizar el 25% de elementos lógicos, 90% de las pines físicos (entradas y salidas del sistema) y un total de memoria utilizada de 31%.

- La arquitectura evaluada es capaz no solo de soportar las exigencias del protocolo Modbus, sino también ejecutar algoritmos adicionales ventajosos para el cliente o usuario final, en este caso se contó con rutinas de diagnóstico del rendimiento instantáneo y acumulado, mediante el uso de estadísticas con las variables de éxito y fracaso de tramas. Algoritmos adicionales para control local o rutinas de seguridad también pueden ser vinculadas.

Desde el punto de vista de las comunicaciones industriales, es muy posible obtener muchos beneficios con en el uso de topologías y arquitecturas como estas, a continuación postularemos algunas de las conclusiones apoyadas en de la experimentación en comunicación realizada:

- El uso de Técnicas de modulación como la DSSS y FHSS, proporcionan seguridad en confiabilidad en la comunicación inalámbrica, pero no se debe dejar de lado la debida utilización de estas, las cuales depende del ambiente y la aplicación en la que se le desee vincular, para el caso de nuestra experimentación en donde se anunció cierto grado de interferencia (dentro de planta), la mejor opción es el uso de la técnica FHSS, la cual preserva de mejor manera los datos sacrificando un poco el ancho de banda de la comunicación, cabe anotar que para el caso estudiado esta última no fue una limitante.
- Es muy importante hacer mucho hincapié en las necesidades del diseño de la comunicación, en función de la confiabilidad y robustez deseada del sistema, muchas normas estudian las mínimas condiciones permisibles para el éxito de las

mismas, tales como ISA, IEEE, FCC entre otras, ya han estudiado en detalle sus modalidades, por lo cual no se debe desconocer estos avances, si en realidad se desea la eficiencia y confiabilidad del sistema.

- Dadas las múltiples situaciones que se pueden representar en aplicación con comunicación inalámbrica, en algunas ocasiones es necesario categorizar el enlace según algunos conceptos tales como: tipo de criticidad en su uso (modo de trabajo o situaciones en la que se puede incurrir en caso de su fallo) y el entorno en que se encuentre el sistema a comunicar, es decir aspectos como son las señales de interferencia en los dos tipos de banda. ISA 100 [22] propone una clasificación según las anteriores condiciones, con la cual nuestra implementación la cual solo está sujeta al monitoreo como clase 5, puede sobrevivir cómodamente en ambientes de baja y moderada contaminación, para ambientes con alta contaminación es muy posible que precise algunos cambios en función de reforzar su inmunidad.
- Como resultado de la implementación y posteriores pruebas se pudo conocer que la transmisión de datos con modulación FHSS es una buena solución para los casos con interferencia moderada y alta, en pruebas realizadas en el interior de planta y en exterior se pudo observar que en distancias con línea de vista menores o iguales a 300 metros es indiferente la contaminación del ambiente industrial, dado que las curvas de prueba muestran 100% éxitos en aciertos tanto en exterior (sin mucha presencia de interferencias) como en interior a dicha distancia. Cabe anotar la distancia enunciada es más que suficiente para la gran mayoría de las aplicaciones

industriales con automatización de procesos en donde se desee hacer implementaciones en comunicación Industrial. La anterior conclusión se soporta bajo el marco de aplicativos de redes industriales inalámbricas WLAN, con el uso de normativas tales como IEEE802.11 o ISA100, las cuales contemplan en la mayoría de sus usos, distancias que varían entre los 30 a 100 metros, uno de los sistemas con múltiples implementaciones es la IWLAN la cual cumple con el estándar IEEE802.11 con distancias no mayores de 100 metros típicamente [12].

- Por fortuna, hay muchas aplicaciones industriales donde los enlaces inalámbricos se pueden implementar casi sin pensar en la posibilidad de interferencias constantes o aleatorias, inclusive aplicaciones que a pesar de existir interferencias estas son relativamente bajas para causar un impacto importante en las comunicaciones inalámbricas, por lo tanto no se precisan modificaciones especiales a la arquitectura o protocolos implementados en dichos casos. Sin embargo existe otra gama de situaciones donde si puede ser necesario el uso de medidas en pro de disminuir el impacto de las interferencias en aplicativos con uso de enlaces inalámbricos en la banda ISM, por lo cual se propone no solo el uso de modulaciones como FHSS, sino también cambios en la potencia de transmisión-recepción, frecuencias en uso, canales de transmisión, ubicación de antenas e implementación nuevos protocolos más seguros que ofrezcan reconocimiento y detección de errores con re-envíos.

A continuación se describen algunos de los principales productos de esta investigación:

- Desarrollo de una plataforma de experimentaciones en comunicación industrial bajo el uso de propagación radio con modulación FHSS, dicha plataforma se presta para continuar con experimentaciones no solo en protocolo Modbus sino también en otros protocolos igualmente usados en la industria.
- Experimentaciones sobre diversas rutinas de protocolo de comunicación Modbus RTU desarrolladas en FPGA de Altera usando uso de un procesador NIOS II en una tarjeta DE1.
- Algoritmo en ANSI C para protocolo Modbus RTU, lenguaje basado en "C" para plataformas conformadas en FPGA y previamente configurado con procesador NIOS II.
- Caracterización de un enlace radio con modulación FHSS, para el envío y recepción de datos de proceso dentro de un ambiente industrial con ruido eléctrico moderado y alto.

Como trabajo siguiente se propone mejorar:

- Optimización: Revisión de la configuración y/o programación del sistema configurado, a pesar de obtener éxito en la comunicación y desempeño del sistema, se propone llevar a cabo una revisión en las sentencias para aumentar el rendimiento y disminuir la cantidad de recursos actualmente utilizados.
- Implementación de lógica interna: en muchos procesos se precisa contar con una lógica interna en el dispositivo de campo que facilite en cierta medida de la capacidad de automatismo a un proceso local, por lo tanto se sigue considerando en dotar FPGA de bloques adicionales para aplicación en campo,

como puede ser el llenado de tanque de cierta solución o un contralor PID para el control de lazo cerrado controlador de Flujo, como también simplemente la lógica de llevar a un punto seguro el sistema bajo la presencia de: una variable salida de rango normal de funcionamiento o realizar un paro del sistema cuando la comunicación presenta muchos errores o salido de funcionamiento.

- Variaciones en el protocolo usado: Se propone adicionar algunas condiciones que mejoren el rendimiento de los enlaces e integridad de los datos, como el caso de lograr un ligero cambio en los algoritmos para permitir que todo esclavo pueda tener una matriz de variables de otros esclavos proveniente de escuchar otros siendo interrogados por el maestro, logrando con esto que el esclavo que escuchó la trama de otro pueda tener conocimiento de esas variables (en caso de necesitarlas) sin que el maestro deba escribirle el dato tiempo después, con esto se ahorraría tiempo en ciclo de comunicación maestro con esclavos.
- Adoptar tendencias actuales de IWLAN: En la actualidad existe una creciente disposición a mejoras de las redes con el uso de Industrial Wireless LAN, la cual ofrece muchas ventajas en acoplamiento y variedad de servicios ya disponibles, por lo tanto se propone ampliar mas esta idea mediante estudio de la norma IEEE802, con sus diferentes adiciones en seguridad y calidad de servicio.

8. Referencias

1. Msc. Jorge Duque. "Procesador de comunicación Modbus. Implementación con Microcontrolador". Tesis de Maestría, Universidad de Santander, Bucaramanga-Colombia. 2005
2. MaxStream. "XStream-PKG-R™ Stand-Alone Radio Modem Operation Manual". 2003
3. Aquilino Rodriguez Penin. "Comunicaciones Industriales, guía práctica". Marcombo. Barcelona-España. Mayo 2008.
4. Modbus-IDA Organization. "MODBUS over serial line. Specification and Implementation guide V1.02". USA. 2006.
5. International Organization for Standardization, International Electrotechnical Commission. "ISO/IEC 7498-1 Open System Interconnection" Second Edition. Corrected and reprinted 1996.
6. American National Standards Institute, Instrumentation Society of America, "ANSI/ISA-50.00.01-1975 (R2002) Compatibility of Analog Signals for Electronic Industrial Process Instruments", "ISA-TR50.02, Parts 3 & 4-2000 Fieldbus Standard for Use in Industrial Control Systems, Parts 3 & 4", "ISA-TR50.02, Part 9, 2000 Fieldbus Standard for Use in Industrial Control Systems: User Layer Technical Report", USA, 1975-2002.

7. Modbus-IDA Organization. "Modbus Application Protocol, V1.1b". 2006. www.modbus.org , descargado: 10-09-2009
8. Laurence Picci, Patrick Salaün. "FPGA-based module: a cost-effective alternative to a PLC solution for I&C system modernization", ISA—The Instrumentation, Systems and Automation Society. Presented at the 14th Annual Joint www.isa.org , 2004. descargado 16-07-2010
9. O. Nuñez, J. Barrero y J. Meneses, "Sistemas de comunicación a nivel industrial " , Universidad de Bucaramanga, 2002.
10. G. Alvarez, J. Pico, L. Vicario, C. Amuchastegui y Ayuso N, "Desarrollo de un protocolo para comunicación Industrial e implementaciones en dispositivos de lógica re-configurable. Facultad de informática, San Sebastián, España. 1999.
11. ISA, "IEC 61158 Technology comparision, State of the bus", Presented at ISA EXPO 2008, 14-16 October 2008, Reliant Center, Houston, Texas.
12. Vicente Gerrero, Ramón L. Yuste, Luis Martinez, "Comunicaciones Industriales", Alfaomega grupo editor S.A, México 2010.
13. G. Prieto, A. Garcia, "Diseño de una Red Inalámbrica IEEE 802.11 FH/CDMA con Protocolo IP para Monitoreo y Control" Universidad de los Andes, Colombia, 2004.
14. L. Herrera, "Telemática y tele- gestión en procesos industriales mediante canales inalámbricos Wi Fi utilizando instrumentación virtual y dispositivos PDA" Grupo de Investigación en Teleinformática de la Universidad Nacional de Colombia, Bogotá, Colombia.

15. Siemens AG, "SimaticNet, Industrial Ethernet, soluciones de red", Industrial automation System, 2006, www.siemens.com/Simatic-net, descargado: 10-06-2010.
16. Saira Mónica Ballesteros, "Redes industriales". Ingeniería Telemática. Universidad católica de Manizales. 2002.
17. ZUO Hong-fei, JIE Xin-chun, WANG Zhi-chun. Based on FPGA Realizing Modbus Communication Protocol, Institute of Information and Engineering, Inner Mongolia University, China, 2007, descargado: 10-09-2010
18. Patrick Salaün, Frederic Daumas, Thuy NGuyen, ISA conference, Why FPGA/ASIC is considered as a key technology for the future of I&C in the power industry at EDF? 2008, descargado 29-09-2010
19. Dick Caro, Wireless Networks for Industrial Automation Wireless systems for industrial automation: Process control and related applications, Second Edition, ISA, USA, 2005.
20. Marko Paavola, Wireless Technologies in Process Automation – Review and an application Example, Report A No 33, University of Oulu, Control Engineering Laboratory, December 2007, 10-10-2010.
21. Low, K.S., W.N.N Win, and J.E. Meng (2005). Wireless Sensor Networks for Industrial Environments. In: Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, IEEE, 2005, 27-10-2010.

22. ISA, ISA 100. 00.01, The Automation Engineer's Guide to Wireless Technology Part 1 : The Physics of Radio, a Tutorial, North Carolina, 2006, 10-03-2010.
23. Altera, User Manual DE1: Development and Education Board, version 1.1, Altera Corporation 2006.
24. ISA-100.11a-2009 .Wireless systems for industrial automation: Process control and related applications, descargado: 30-10-2010
25. Altera Corporation, Introduction to the Altera SOPC Builder, Using VHDL Design, 2008, www.altera.com, descargado: 02-03-2010.
26. Simatic HMI, ProTool/Pro Runtime User's Guide, Siemens Alemania, 2001, descargado: 10-08-2009.
27. Modbus-IDA Organization. "MODBUS over serial line specification and implementation guide V1.02". 2006. www.modbus.org , descargado: 10-09-2009.
28. Ivan Howitt, Wayne W. Manges, Phani Teja Kuruganti, Glenn Allgood, José A. Gutierrez, James M. Conrada, "Wireless industrial sensor networks: Framework for QoS assessment and QoS management.", ISA Transactions Volume 45, Number 3, July 2006, descargado: 11-08-2010.

9. Anexos

9.1. Programación en NIOS II de Altera.