

Diseño e Implementación de un Transmisor Digital Inalámbrico mediante Tecnología FPGA

Reyes Cruz Luis Ángel, Pedraza Ortega Jesús Carlos, Ramos Arreguín Juan Manuel,
Díaz Delgado Guillermo, Tovar Arriaga Saúl.

Facultad de Informática-Facultad de Ingeniería, Universidad Autónoma de Querétaro.
Av. de las Ciencias S/N Campus Juriquilla, Juriquilla, Querétaro, Qro.
C.P. 76230 México. Tel. 1921200 EXT. 5900
reyes.luis1010@gmail.com

Resumen

En este trabajo se presenta el diseño e implementación de un transmisor digital inalámbrico en una tarjeta de desarrollo con tecnología de lógica programable de tipo FPGA (Field Programmable Gate Array). El transmisor emplea un esquema de modulación por desplazamiento binario de fase (Binary Phase Shift Keying, BPSK), e incluye también un codificador de canal basado en el código corrector de Hamming H(7,4).

La programación de los bloques funcionales del transmisor se realizó mediante el lenguaje de programación de hardware VHDL (Very High Speed circuit Description Language). El transmisor fue validado mediante pruebas de simulación en Simulink antes de ser implementado en el dispositivo semiconductor FPGA para las pruebas finales de funcionamiento.

Palabras clave: transmisor digital inalámbrico, BPSK, VHDL, Hamming FPGA.

1. Introducción

En la actualidad los avances tecnológicos han tenido un gran efecto en la sociedad al grado de que en nuestra vida cotidiana ya no podemos prescindir de servicios como las comunicaciones (telefonía, internet, televisión, radio, etc.). La evolución de las comunicaciones nos permite como usuarios tener una variedad de opciones de compañías que ofrecen los servicios, reduciendo los costos y teniendo un notable aumento en la calidad del servicio. Estos avances se deben en gran parte al desarrollo que la tecnología electrónica ha tenido y en especial la electrónica digital, de manera que se están migrando los servicios de tecnología analógica a tecnología digital, debido a las grandes ventajas que poseen las comunicaciones digitales como: un menor ancho de banda necesario, mayor calidad en el servicio, mejora en la utilización de la banda de frecuencias, facilidad de procesamiento, facilidad de multiplexado e inmunidad al ruido [1].

El objetivo básico de un sistema de comunicación es la transferencia de información de un punto geográfico a otro. En general, los sistemas de comunicación (SDC) se componen de tres subsistemas principales: el transmisor, el canal de comunicación y el receptor.

Los sistemas tradicionales de comunicación que usan esquemas de modulación analógica, como la modulación de amplitud (AM), modulación de frecuencia (FM) y la modulación de fase (PM) se están sustituyendo rápidamente debido al gran avance de la electrónica digital y, por consiguiente, el desarrollo de sistemas de comunicaciones digitales, los cuales poseen ventajas sobre los sistemas analógicos que permiten la optimización del canal de transmisión. Así pues, los SDC facilitan el

procesamiento de datos y señales, proporcionando una mayor capacidad de multiplexación de comunicaciones simultáneas, así como una mayor inmunidad al ruido y a la interferencia. Los sistemas digitales de comunicación se encuentran conformados por portadoras de frecuencias relativamente altas que se modulan mediante señales de información digital de relativamente bajas frecuencias [2]. La figura 1 muestra un diagrama general a bloques de un sistema digital de comunicación.

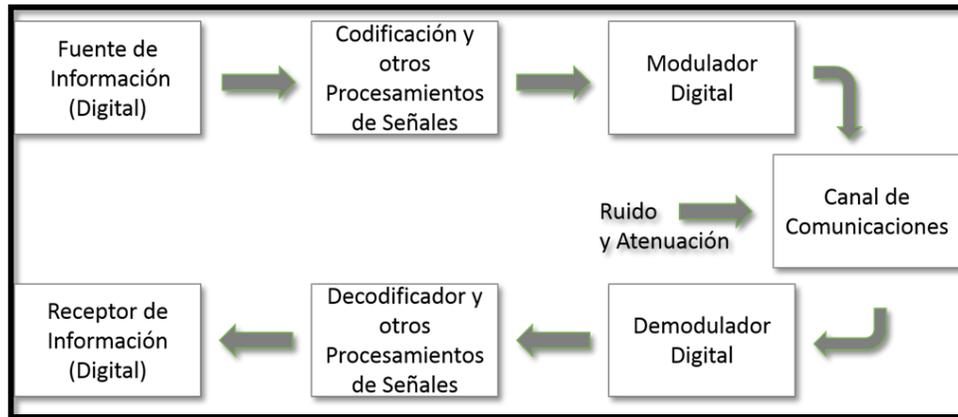


Fig. 1. Diagrama general de bloques de un sistema digital de comunicación [3].

Por otra parte, una de las herramientas más poderosas con las que se cuenta para el diseño de sistemas digitales son los Lenguajes Descriptivos de Hardware (LDH) que permiten realizar una estructura jerárquica de las etapas del diseño logrando una optimización en el trabajo.

VHDL (Very High-speed Integrated Circuit Hardware Description Language) es el lenguaje descriptivo más usado en la actualidad. Su popularidad se debe principalmente a que es un estándar del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, Institute of Electrical and Electronics Engineers). El modelado de circuitos en VHDL se basa principalmente en la definición de arquitecturas que representan cada una de las estructuras digitales que pueden aplicarse al diseño del circuito. A partir de cada una de estas estructuras, se pueden declarar entidades con sus correspondientes elementos para llevar a cabo la simulación del circuito y la implementación en dispositivos lógicos programables [4].

Los lenguajes descriptivos de hardware sirven como una herramienta para obtener soluciones a problemas de diseño de manera rápida y eficiente. Se han desarrollado componentes de hardware que permiten la programación mediante estos lenguajes descriptivos. Estos componentes van desde pequeños dispositivos, con unas cuantas compuertas lógicas, hasta dispositivos más complejos, con una capacidad de integración de miles de compuertas agrupadas en bloques de funcionales [5].

Algunos trabajos sobre sistemas de comunicaciones implementados en Dispositivos Lógicos Programables (PLD's Programmable Logic Device) han probado la eficiencia en su implementación gracias a la arquitectura de los PLD's que permiten: facilidad y rapidez de diseño, flexibilidad y portabilidad; capaz de funcionar de manera correcta en distintas áreas de aplicación como: aplicaciones RFID como un transmisor [6]; en redes inalámbricas mediante la creación de un sistemas en FPGA's usando esquemas como OFDM y modulación BPSK para la implementación del estándar IEEE 802.11 [7]; en sistemas de comunicaciones de alta velocidad implementándose en arquitecturas de FPGA especiales para telecomunicaciones [8]; en biomedicina como el diseño de un modulador para telemetría biomédica programada en VHDL, los cuales tienen varias aplicaciones en la ésta área como los dispositivos transcutáneos que contienen sensores con tecnología similar a la de los sistemas RFID[9].

2. Metodología

El modelado de circuitos digitales en VHDL consta básicamente de cinco pasos fundamentales para su desarrollo, como se puede observar en el diagrama de la Figura 2.

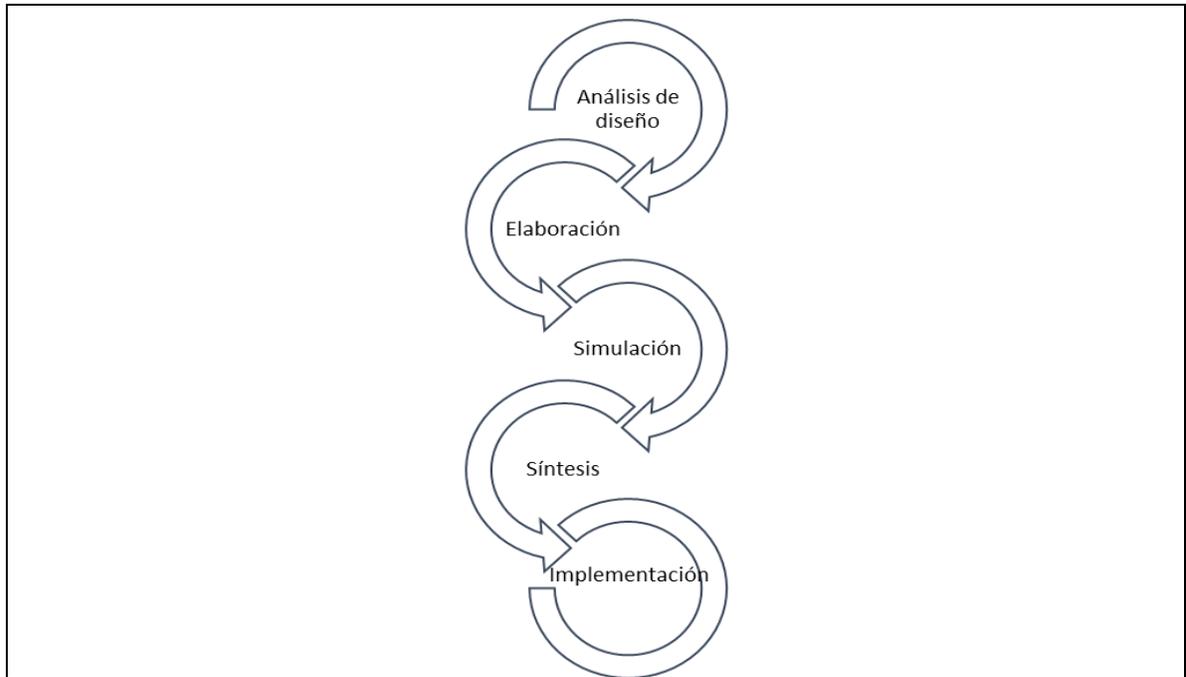


Fig. 2. Metodología para el modelado de circuitos digitales.

La fase inicial del modelado consiste en el análisis del diseño a partir de los requerimientos, características del sistema y del circuito a implementar. La elaboración del modelo se realiza una vez que se han detectado y corregido los errores en el diseño, entonces se continúa con la especificación del diseño en VHDL. Al término de la fase de elaboración se procede a ejecutar pruebas de simulación del modelo mediante la configuración de las señales de entrada y los tiempos de la simulación, en ésta parte es recomendable la creación de bancos de pruebas para una evaluación óptima de la simulación.

La síntesis del sistema se realiza mediante una herramienta de síntesis, con la cual es posible convertir el diseño a una abstracción de bajo nivel. Una vez que se realiza la síntesis de manera correcta, es factible la implementación del sistema en una tarjeta de desarrollo para realizar las pruebas pertinentes [10].

2.1 Análisis de diseño

El diagrama de bloques que conforma nuestro transmisor digital inalámbrico se desarrolla a partir de la Figura 1, tomando solamente la parte del transmisor y declarando cada uno de los bloques a implementar. En la Figura 3 se muestran cada uno de los bloques que conforman el transmisor digital inalámbrico propuesto.

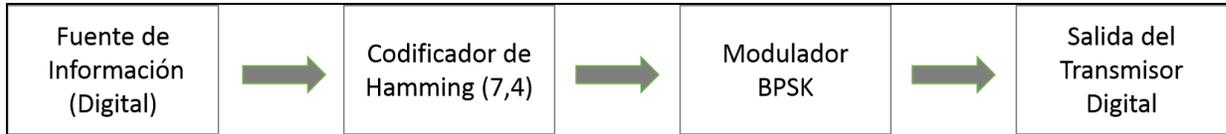


Fig. 3. Diagrama a bloques del transmisor digital de datos.

La fuente de información, puede ser cualquier señal binaria aleatoria; generada arbitrariamente o mediante un banco de pruebas.

El codificador de Hamming implementado es un código (7,4), de manera que, si queremos transmitir la palabra “1100”, el desarrollo del código de Hamming será el que se muestra en la Tabla 1. Los dígitos de comprobación se encuentran ubicados en las posiciones binarias donde solo hay un ‘1’, que corresponden a las posiciones 1, 2 y 4 en binario. Las otras posiciones se llenan con los dígitos del mensaje. Para el cálculo de los dígitos de comprobación, se utiliza la relación de la Tabla 1 para la obtención de las ecuaciones. Los dígitos de mensaje que se encuentran en cada ecuación de comprobación corresponden a las posiciones binarias que contienen un ‘1’ en la misma posición que la del dígito de comprobación a calcular [11].

Tabla 1. Desarrollo del código de Hamming (7,4).

Palabra	c_1	c_2	1	c_3	1	0	0
Posición binaria	001	010	011	100	101	110	111
Tipo de dígito	c_1	c_2	m_1	c_3	m_2	m_3	m_4
c_1	0		1		1		0
c_2		1	1			0	0
c_3				1	1	0	0
Palabra codificada	0	1	1	1	1	0	0

Así las ecuaciones para los bits de comprobación quedan de la siguiente manera:

$$c_1 = 1 \oplus 1 \oplus 0 = 0 \quad (1)$$

$$c_2 = 1 \oplus 0 \oplus 0 = 1 \quad (2)$$

$$c_3 = 1 \oplus 0 \oplus 0 = 1 \quad (3)$$

Obteniendo la palabra codificada “0111100”, donde el bit menos significativo o el primer bit a transmitir corresponde al bit de la izquierda. Posteriormente esta palabra codificada será el conjunto de bits que entren al modulador.

Para el modulador BPSK, se tomaron en cuenta las ecuaciones especificadas anteriormente, se consideró una señal portadora senoidal con una frecuencia en el orden de los KHz, esto debido a los ciclos de reloj utilizados para generar la señal e implementar la comunicación con el DAC. La estructura del modulador se describe en la Figura 4.

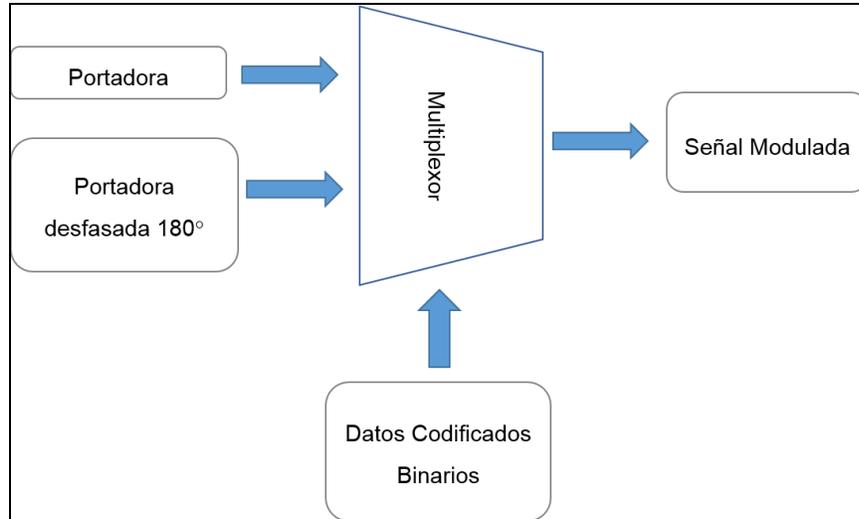


Fig. 4. Diagrama del Modulador BPSK.

2.2 Elaboración.

El sistema se diseñó con un enfoque “top-down” para la generación de los bloques funcionales en VHDL; se utilizó el editor de textos de Active-HDL para este proceso. La entidad general del transmisor digital de datos a implementar en Hardware se muestra en la Figura 5. Las señales de entrada son: el reloj maestro del sistema (CLK), la señal de RST para resetear las señales que dependen del reloj maestro. Y la señal INF la cual contiene el grupo de los bits a transmitir sin codificar. Las señales de salida son: las cuatro señales para el DAC y la señal modulante para su visualización física.

La primera parte del desarrollo, consistió en el diseño de la portadora y la portadora desfasa, para lo que se utilizó el código de Matlab de Rene Troncoso [1] para la generación de una Look Up Table de 12 bits correspondiente a una función senoidal y que se binariza en 4096 valores. De esta manera se puede implementar una aproximación de la función seno en una FPGA. La entidad de LUT de la Portadora consiste en una señal de entrada de 12 bits que se mapea a la salida con su correspondiente valor de la función senoidal. La entidad de la LUT para la portadora desfasada es similar, solo que la función mapeada corresponde a la función senoidal desfasada 180°.

EL bloque del Acumulador de fase realiza una suma sistemática de fase, acumulándola de manera que sirva para desplazarse en los valores de las LUT's de una forma cíclica. La entidad del Acumulador de fase se muestra en la Figura 6. Las señales de entrada son: el reloj maestro del sistema (CLK), la señal de RST para resetear las señales que dependen del reloj maestro. Y la señal P de 12 bits la cual establece el valor de suma para el acumulador. Las señal de salida e es la suma acumulada de 12 bits proveniente del registro.

El Acumulador de fase consiste en un sumador de 12 bits, una unidad de registro de 12 bits para almacenar la suma acumulada y una unidad de registro de 1 bit para almacenar el acarreo de la sumatoria.

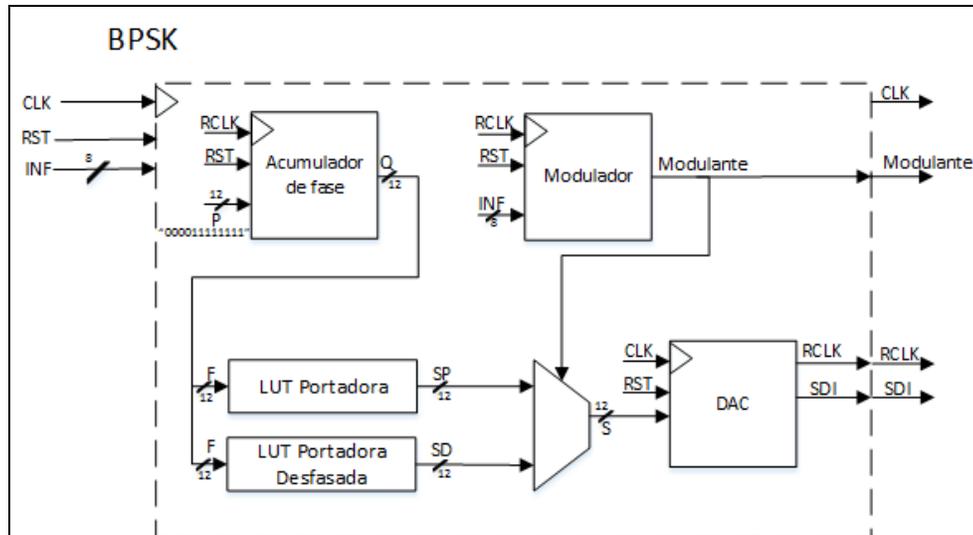


Fig. 5. Entidad general del transmisor digital de datos.

En el sumador de 12 bits, las señales de entrada son: El valor establecido para la sumatoria R1, el valor de la sumatoria acumulada R2 y el acarreo de entrada correspondiente a la salida del registro de acarreo. Mientras que el resultado de la sumatoria se pasa a la salida del sumador. La unidad de registro de 12 bits almacena la sumatoria acumulada y se actualiza en cada ciclo de reloj con respecto a la salida del sumador. Por lo que sus señales de entrada correspondientes son: el reloj maestro, el reset y el resultado del sumador. A la salida se encuentra la señal de la sumatoria acumulada.

En el bloque de acarreo, se almacena el décimo tercer bit de la sumatoria y pasa como entrada en el sumador. El valor de almacenamiento se actualiza en cada ciclo de reloj igual que la unidad de registro de 12 bits.

En el bloque del Modulador consiste en un divisor de frecuencia, el codificador de Hamming y un buffer para separar los datos codificados y mostrarlos en la salida. La entidad del Modulador se muestra en la Figura 7. Las señales de entrada son las mismas que las de la entidad general BPSK, y la señal de salida es la señal binaria codificada.

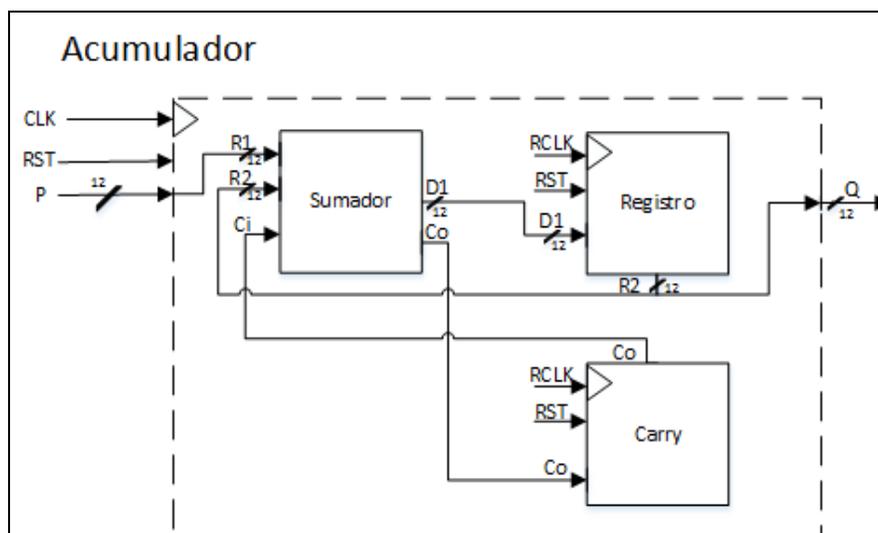


Fig. 6. Entidad del Acumulador de fase.

El divisor de frecuencia se establece de acuerdo a la velocidad de bits requerida para la transmisión mediante la siguiente fórmula, donde M es el valor que debe ser programado para la disminución de la frecuencia de reloj:

$$M = \frac{\text{Periodo deseado}}{\text{Periodo de reloj}} \quad (4)$$

De esta manera la salida del divisor de frecuencia sirve para que el buffer entregue a su salida un bit a la vez proveniente del codificador. El codificador recibe en su entrada el grupo de 4 bits que codifica de acuerdo con las ecuaciones para los bits de comprobación establecidas anteriormente.

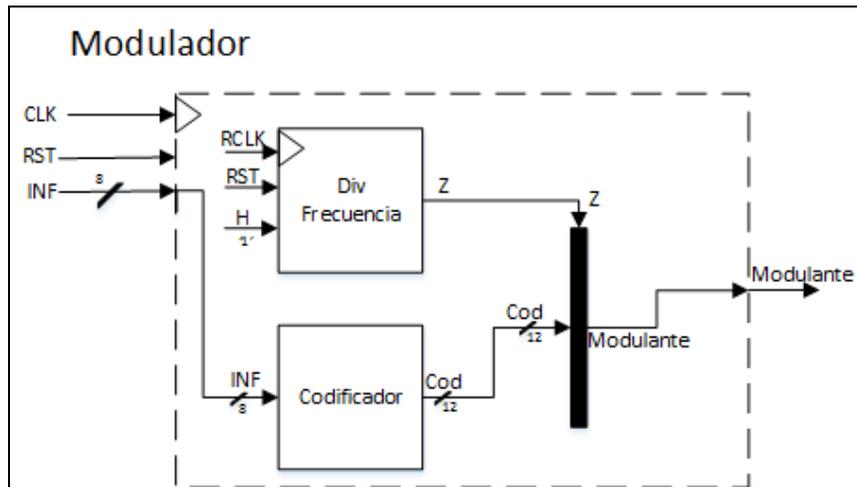


Fig. 7. Entidad del Modulador.

Las salidas de las LUT's entran a un multiplexor 2 a 1 y la señal selectora proviene de la salida del bloque del modulador, de esta manera, si la señal selectora es un '0' lógico, la salida del multiplexor entrega la señal de la portadora, y si la señal selectora es un '1' lógico, la salida del multiplexor contendrá la señal desfasada de la portadora. La salida del multiplexor está formada por 12 bits, los cuales entran al bloque del DAC.

El bloque del DAC contiene una máquina de estados finita para la implementación del protocolo de 24 bits del DAC LTC2624 [12]. El esquema de la máquina de datos implementada se muestra en la Figura 8.

En este esquema se puede observar las señales que interactuarán con la máquina de estados (SCK, CS, MOSI, RDY e Index), la señal SCK funciona como el reloj para la sincronización con el DAC, la señal CS funciona como reset para la comunicación del DAC, la señal MOSI toma el valor de cada uno de los datos del protocolo de 24 bits. La señal RDY funciona para indicar que los 24 bits fueron transmitidos y la señal de index funciona para el desplazamiento de los 24 bits del protocolo. En total son 25 ciclos los que se requieren para completar el protocolo.

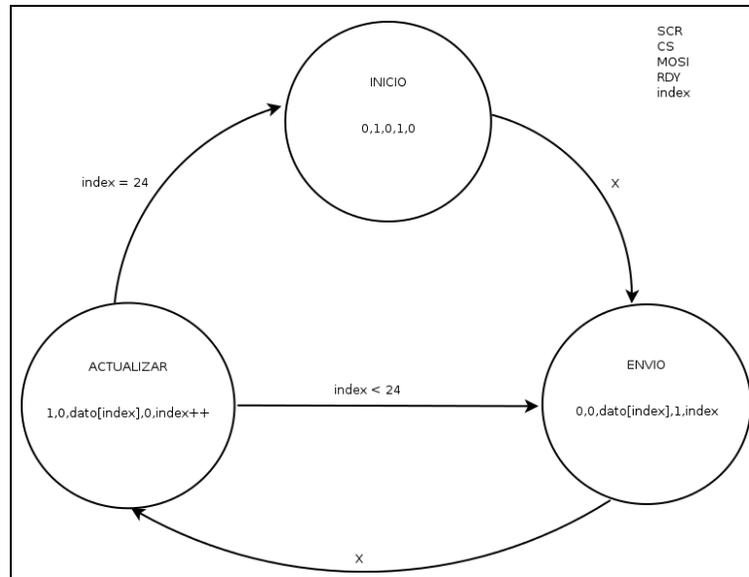


Fig. 8. Esquema de la Máquina de Estados Finita para el DAC.

2.3 Simulación.

Teniendo el desarrollo en VHDL de las entidades descritas anteriormente, se realizó una simulación en Active-HDL mediante el Visor de Formas de Onda, para la validación del transmisor digital.

La simulación en XSG nos ayuda a tener otro método para la validación del código VHDL generado, este método genera un entorno más similar a lo que sería la implementación en hardware ya que se simula también la capacidad de la tarjeta de desarrollo utilizada, por lo que si un diseño sobrepasa la capacidad de la tarjeta, la simulación no funcionará [13].

2.4 Síntesis.

Para la síntesis del código VHDL se usa Xilinx ISE, por lo que se exportó el código generado en Active-HDL y se importó en Xilinx ISE. Un paso previo a la síntesis es la asignación de los pines de la FPGA con su correspondiente señal del transmisor, usando la herramienta PlanAhead y la tabla 2 que indica la señal y el pin utilizado de la FPGA.

Una vez que se tiene la asignación de los pines, se realiza la síntesis para la abstracción a bajo nivel del código VHDL, esto da como resultado la generación del archivo de programación (.bit) que se utiliza para la implementación en el hardware. El archivo (.bit) es creado con el mismo nombre del archivo VHDL de más alto nivel y en el mismo directorio en el que se encuentra el proyecto creado en Xilinx ISE.

Tabla 2. Tabla para la asignación de pines en la FPGA.

<i>Señal</i>	Tipo de señal (entrada o salida)	Numero de Pin en la FPGA
AD_CONV	Salida	P11
AMP_CS	Salida	N7
FPGA_INIT_B	Salida	T3
SF_CEO	Salida	D13
SPI_SS_B	Salida	U3
CLK	Entrada	C9
RST	Entrada	L3
Modu	Salida	D7
Enc	Salida	F12
DAC_CLR	Salida	P8
DAC_CS	Salida	N8
DAC_MOSI	Salida	T4
DAC_SCK	Salida	U16

2.5 Implementación.

Para la programación del (.bit) en la tarjeta de desarrollo, se utilizó la herramienta de iMPACT. Ya que se tiene conectada correctamente la tarjeta mediante el puerto USB de programación, iMPACT escanea la tarjeta y genera la cadena de elementos programables, posteriormente se selecciona el archivo de programación para cargarlo a la tarjeta de desarrollo.

La tarjeta de desarrollo empleada es la Spartan-3E Starter Kit de Xilinx [14]. La conexión de la tarjeta de desarrollo y el osciloscopio digital se llevó a cabo utilizando dos puntas para osciloscopio. La primera punta conectada a la salida del canal A del DAC, usando la tierra que contiene el conector de 6 pines del DAC. Esta punta contendrá la señal modulada.

La segunda punta servirá para visualizar la señal de información, por lo que se conecta en el pin D7 de la tarjeta como se especificó en la Tabla 2. La tierra para esta punta puede ser la misma que para la primera punta.

La Figura 9 muestra la conexión física de la tarjeta de desarrollo y el osciloscopio, donde se puede observar las dos puntas del osciloscopio y el cable de alimentación de la tarjeta.

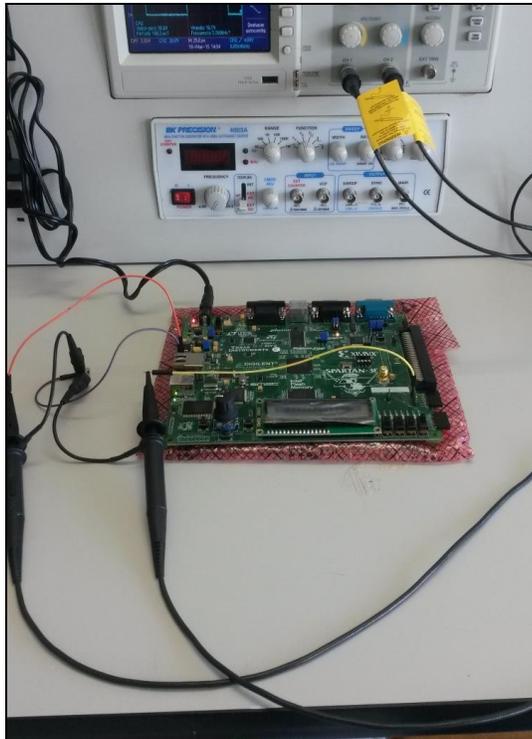


Fig. 9. Conexión de la tarjeta de desarrollo y el osciloscopio digital.

3. Análisis de resultados

Los resultados de la simulación en Active-HDL se muestran en la Figura 10. Se observa la señal generada por LUT de la portadora y de la portadora desfasada. La señal modulada realiza los cambios de fase de acuerdo a la señal de información o modulante. La frecuencia de la portadora está limitada ya que con un reloj maestro de 50MHz, el DAC puede trabajar hasta 25MHz. Ya que el protocolo para la comunicación consta de 25 estados, tenemos que la frecuencia de actualización del DAC es de 1MHz. Por lo que cada cambio en la salida del DAC se da a 1ns. Si definimos una portadora senoidal en 16 valores discretos junto con la frecuencia de actualización del DAC, obtenemos una frecuencia de 62.5KHz

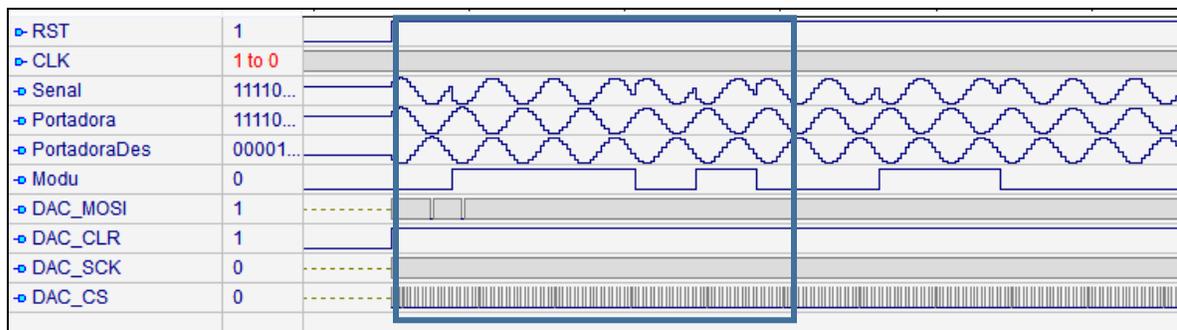


Fig. 10. Simulación en Active-HDL.

En cuanto a la implementación en la tarjeta de desarrollo del transmisor digital incluyendo la codificación y la modulación, la Figura 11 muestra el despliegue de la señal de información (en azul) y la señal modulada (en amarillo) en el osciloscopio.

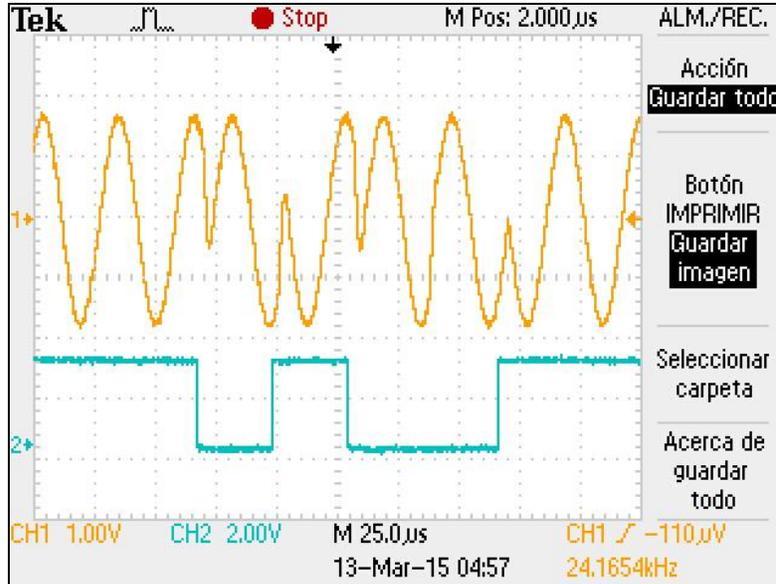


Fig. 11. Señal de información y señal modulada visualizadas en el osciloscopio.

Los recursos utilizados de la tarjeta de desarrollo para la implementación del transmisor digital se muestran en la Tabla 3. Se puede apreciar que el uso de los recursos de la tarjeta son menores al 70%, por lo que se es capaz de agregar módulos adicionales.

Tabla 3. Utilización de recursos de la tarjeta de desarrollo.

Recursos	Usados	Porcentaje Utilizado
Flip-Flops	54	1%
LUT de 4 entradas	2494	26%
Segmentos Ocupados	1479	31%
Buffer de Entrada/Salida	13	5%
Buffers de Reloj	2	8%

4. Conclusiones

En éste trabajo se presentó el desarrollo e implementación de un transmisor digital de datos utilizando un esquema de codificación para el control de errores mediante códigos de Hamming y un esquema de modulación BPSK. Se llevó a cabo un análisis de los sistemas de comunicaciones digitales, de los lenguajes descriptivos de hardware y de la tecnología FPGA para la definición de la metodología que abarcara el diseño completo del transmisor. La validación del diseño del transmisor se hizo por medio de simulaciones que comprobaran el funcionamiento antes de llevarlo a la etapa de implementación para optimizar los tiempos de desarrollo. La implementación se realizó en una tarjeta de desarrollo Spartan-3E Starter Kit que contiene un DAC para la salida de la señal modulada.

Por medio de este trabajo se pudo comprobar que el desarrollo de sistemas de comunicaciones mediante tecnología FPGA es factible y que se pueden explotar los beneficios que ofrece, en especial la flexibilidad de diseño y el tiempo de desarrollo, gracias al concepto modular con el que se realiza la programación. En cuanto al funcionamiento del sistema en la tarjeta de desarrollo, los resultados son los esperados y la respuesta de la tarjeta comprueba que el paralelismo puro de las FPGA's da una ventaja en el desarrollo de sistemas digitales complejos.

Referencias

- [1] Romero J., "Electrónica digital y lógica programable". Universidad de Guanajuato, México. 1a edición, 2007.
- [2] Wayne T., "Sistemas de comunicaciones electrónicas". Pearson Educación, México. 4a edición, 2003.
- [3] Couch II L., "Sistemas de comunicación digitales y analógicos". Prentice Hall, México. 7a edición, 2002.
- [4] García M. & Pérez J., "Dispositivos lógicos programables (PLDs): diseño práctico de aplicaciones". RA-MA. México. 2006.
- [5] Aceves M. & Ramos J., "Fundamentos de Sistemas Embebidos", Asociación Mexicana de Mecatrónica A.C. México. 1a edición, 2012.
- [6] Van de Beek, R., Ciacci, M., Al-Kadi, G., Kompan, P., & Stark, M. "A 13.56 Mbps PSK receiver for 13.56 MHz RFID applications". In Radio Frequency Integrated Circuits Symposium (RFIC). IEEE (pp. 239-242). 2012.
- [7] Garcia, J., & Cumplido, R. "On the design of an FPGA-Based OFDM modulator for IEEE 802.16-2004". In Reconfigurable Computing and FPGAs, International Conference on (pp. 4-pp). IEEE. 2005.
- [8] Tsutsui, A., Miyazaki, T., Yamada, K., & Ohta, N. "Special purpose FPGA for high-speed digital telecommunication systems". In Computer Design: VLSI in Computers and Processors, Proceedings. IEEE International Conference on (pp. 486-491). 1995.
- [9] Elamary, G., Chester, G., & Neasham, J. "An analysis of wireless inductive coupling for High Data Rate biomedical telemetry using a new VHDL n-PSK modulator". In Electronics, Circuits, and Systems. 16th IEEE International Conference on (pp. 211-214). 2009.
- [10] Sánchez I., García M. Saldívar P., Montiel P., Sepúlveda R., & Charles R., "Modelado de circuitos lógicos digitales en Active HDL 5.1.", VHDL-Encuentro de Investigación en Ingeniería Eléctrica 2006. (pp. 169-174). ENINVIE. 2006.
- [11] Bruce C., Crilly P. & Rutledge J., "Sistemas de comunicación". McGraw-Hill, México. 4a edición, 2002.
- [12] Xilinx, Inc., "Spartan-3E FPGA Starter Kit Board User Guide". 2011.
- [13] Linear, Technology. 2004. LTC2604/LTC2614/LTC2624 datasheet.
- [14] Digilent Inc. 2014.
<http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,897&Prod=NEXYS3&CFID=5644420&CFTOKEN=24ce718231f10800-1F4F5181-5056-0201-020930A9E23F6C12>.