

Uso de un FPGA (Field Programmable Gate Array) para la implementación de la sección de banda base de la capa física de un transmisor basado en el estándar IEEE 802.11n en modo Greenfield

Iván Bernal y Romel Espinosa, Escuela Politécnica Nacional (EPN), Quito-Ecuador

Resumen – En el documento se presentará la descripción de los bloques principales que forman parte de la sección de banda base de la capa física de un transmisor basado en el estándar IEEE 802.11n en modo Greenfield. Además, se incluyen las ideas centrales para la implementación de cada uno de los bloques presentados en el estándar, así como del transmisor completo. Adicionalmente, el estándar IEEE 802.11n proporciona un conjunto de pruebas que sirven como referencia para verificar el funcionamiento de los bloques implementados; por tal motivo, las pruebas realizadas en el presente proyecto consisten principalmente en simulaciones y procedimientos de hardware de co-simulación que fueron realizadas a cada uno de los bloques desarrollados.

Índices – Banda base del estándar IEEE 802.11n, FPGA, Lenguaje VHDL, System Generator.

I. INTRODUCCIÓN

En los últimos años se ha podido observar que el diseño digital utilizando dispositivos lógicos programables como son los FPGAs (Field Programmable Gate Arrays) ha tenido un gran desarrollo debido a las ventajas que éstos ofrecen con respecto a otros dispositivos como son los CPLDs (Complex Programmable Logic Device) o ASICs (Application Specific Integrated Circuits). Por otra parte, el incremento de las velocidades de transmisión y de la cobertura que ofrecen las redes inalámbricas basadas en el estándar IEEE 802.11n está permitiendo que el nuevo estándar sea aceptado rápidamente. En consecuencia, el presente proyecto pretende implementar los principales bloques de la sección de banda base de la capa física de un transmisor basado en el estándar IEEE 802.11n en modo Greenfield utilizando un FPGA.

El presente documento se encuentra estructurado por cinco secciones las cuales fueron seleccionadas para brindar al lector una fácil comprensión de los temas tratados. En primer lugar,

se presenta una breve introducción al documento. A continuación, la segunda sección presenta la descripción de los principales bloques que integran el transmisor. Mientras tanto, la tercera sección está dedicada íntegramente a la implementación de los bloques, comenzando con los requerimientos de diseño, y posteriormente continuando con la presentación de las ideas principales que fueron utilizadas para la implementación de los bloques. En la cuarta sección, se menciona el procedimiento realizado para la ejecución de las pruebas de los bloques individuales, así como del transmisor completo. Y, finalmente, la quinta sección corresponde a las conclusiones que se obtuvieron después de la ejecución del proyecto.

II. DESCRIPCIÓN DE LOS BLOQUES QUE INTEGRAN EL TRANSMISOR

El estándar IEEE 802.11n presenta la descripción del funcionamiento de los bloques que conforman el transmisor, sin precisar la manera cómo realizar la implementación, brindando al fabricante la posibilidad y la libertad de seleccionar el método más conveniente. La Fig. 1 muestra los bloques que forman parte de la sección de banda base que integran el presente proyecto. Adicionalmente, se pueden incrementar algunos bloques extras a la Fig. 1 para ofrecer características MIMO complementarias, las mismas que fueron consideradas como opcionales en el proceso de aprobación del estándar.

A. Bloque Aleatorizador [1], [2]

Es el encargado de aleatorizar los bits de datos que ingresan en el transmisor, con el fin de evitar las largas secuencias de ceros o unos. El bloque se basa en el polinomio generador $S(x)$ (1).

$$S(x) = x^7 + x^4 + 1 \quad (1)$$

Los valores iniciales utilizados para comenzar con el funcionamiento del bloque Aleatorizador son los siguientes [1 0 1 1 1 0 1] y son denominados valores semilla.

I. Bernal, es profesor principal en la Facultad de Ingeniería Eléctrica y Electrónica de La Escuela Politécnica Nacional, Quito-Ecuador, (e-mail: ivan.bernal@epn.edu.ec).

R. Espinosa, se desempeña como Ingeniero de Radio Frecuencia en Konectado, Quito-Ecuador, (e-mail: romel_ivan333@yahoo.es).

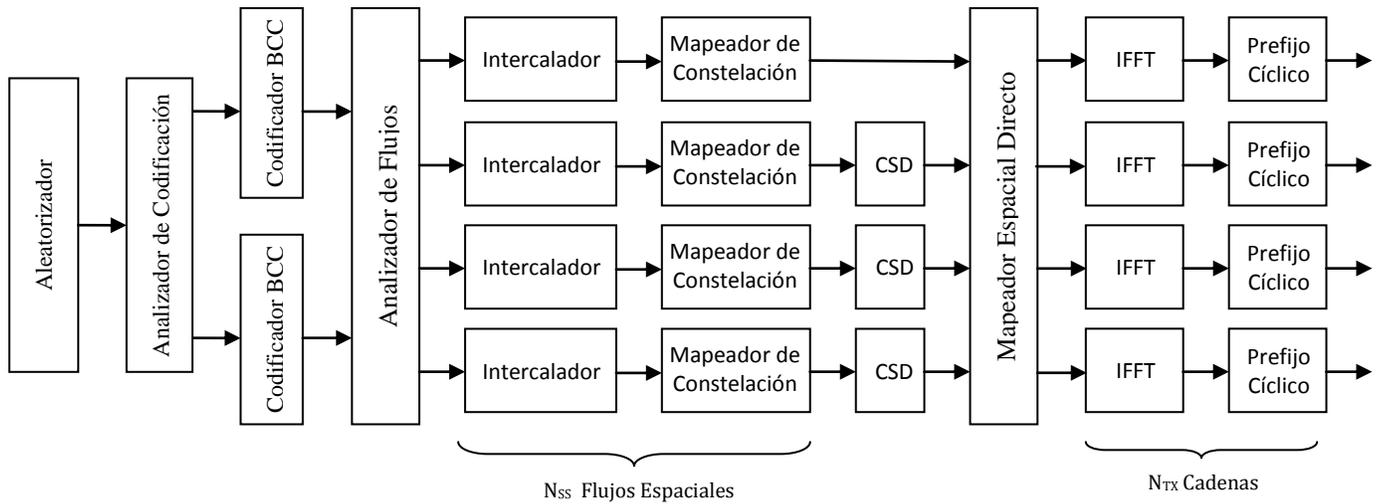


Fig. 1 Diagrama de bloques del transmisor para el campo de datos

B. Bloque Analizador de Codificación [1], [2]

El bloque Analizador de Codificación realiza la distribución de los bits aleatorizados que provienen del bloque Aleatorizador mediante el método de round robin. Este bloque es utilizado cuando se desean alcanzar tasas de transmisión que superen los 300 Mbps.

El método round robin consiste en distribuir de manera alternada los bits de entrada en cada una de las dos salidas disponibles.

C. Bloque Codificador Convolutivo Binario [1], [2]

El bloque Codificador Convolutivo Binario se encarga de realizar la codificación convolutiva de los bits aleatorizados y distribuidos. El bloque utiliza los códigos convolucionales $g_0 = 133_8$ y $g_1 = 171_8$ con tasa $R = 1/2$.

La Fig. 2 muestra el funcionamiento del codificador convolutivo. Adicionalmente, se pueden incorporar patrones de perforado para incrementar las tasas de transmisión. Es así, que el estándar IEEE 802.11n permite cuatro opciones de tasas de codificación las mismas que son presentadas en la Tabla I.

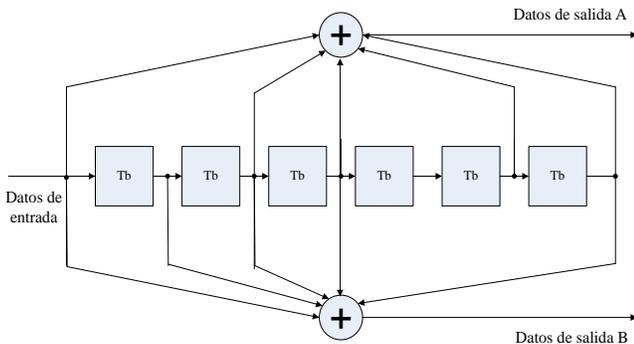


Fig. 2 Codificador convolutivo ($k=7$) [1]

TABLA I
TASAS DE CODIFICACIÓN PARA IEEE 802.11n

	Tasa de Codificación
1	1 / 2
2	2 / 3
3	3 / 4
4	5 / 6

D. Bloque Analizador de Flujos [1], [2]

El bloque Analizador de Flujos se encarga de realizar la distribución de los flujos de bits de datos que provienen de los bloques Codificadores Convolucionales Binarios en los diferentes flujos espaciales. Para el caso que todos los flujos espaciales utilicen el mismo tipo de mapeado se debe utilizar (2) para determinar el número de bits que se asignan a cada flujo espacial. Mientras tanto, si el tipo de mapeado es diferente para los distintos flujos espaciales se utiliza (3) para determinar la cantidad de bits que deben distribuirse en cada flujo espacial.

$$s = \max \left\{ 1, \frac{N_{BPSCS}}{2} \right\} \quad (2)$$

$$s(i_{SS}) = \max \left\{ 1, \frac{N_{BPSCS}(i_{SS})}{2} \right\} \quad (3)$$

donde: N_{BPSCS} corresponde al número de bits por subportadora.

El valor de s o $s(i_{SS})$ corresponde al número de bits que se asignan a cada uno de los ejes en la representación del diagrama de constelación, es decir, el valor de s o $s(i_{SS})$ depende únicamente del tipo de mapeado de constelación seleccionado en cada flujo espacial.

E. Bloque Intercalador [1], [2]

El bloque Intercalador es encargado de cambiar el orden de salida de los bits codificados que ingresan en éste. La finalidad

de este bloque es proporcionar un método que permita reducir el efecto de errores de ráfagas que se producen en los datos transmitidos en canales ruidosos.

El estándar IEEE 802.11n considera tres etapas de permutaciones.

1) *Primera Permutación:* Se encarga que los bits adyacentes sean ubicados en portadoras no cercanas. La ecuación (4) es utilizada para calcular los valores correspondientes a la primera permutación.

$$i = N_{ROW}(k \bmod N_{COL}) + \text{floor}(k/N_{COL}) \quad (4)$$

$$k = 0, 1, \dots, N_{CBPSS}(i_{SS}) - 1$$

donde: N_{ROW} indica el número de filas.
 N_{COL} corresponde al número de columnas.

2) *Segunda Permutación:* Facilita que los bits adyacentes se sitúen de manera alternada entre más y menos significativos. Para el cálculo de los valores se emplea (5).

$$j = s(i_{SS}) \times \text{floor}(i/s(i_{SS})) + (i + N_{CBPSS}(i_{SS}) - \text{floor}(N_{COL} \times i/N_{CBPSS}(i_{SS}))) \bmod s(i_{SS}) \quad (5)$$

$$i = 0, 1, \dots, N_{CBPSS}(i_{SS}) - 1$$

donde: N_{CBPSS} corresponde al número de bits codificados por símbolo.
 $s(i_{SS})$ corresponde al valor calculado en (3).

3) *Tercera Permutación:* Permite que los bits se coloquen con una rotación de frecuencia. La ecuación (6) es utilizada para el cálculo de los valores correspondientes.

$$r = \left[j - \left(((i_{SS} - 1) \times 2) \bmod 3 + 3 \times \text{floor}((i_{SS} - 1)/3) \right) \times N_{ROT} \times N_{BPSCS}(i_{SS}) \right] \bmod N_{CBPSS}(i_{SS}) \quad (6)$$

$$j = 0, 1, \dots, N_{CBPSS}(i_{SS}) - 1, i_{SS} = 1, 2, 3, 4$$

donde: N_{ROT} indica el número de rotación.

F. Bloque Mapeador de Constelación [1], [2]

El bloque Mapeador de Constelación es el encargado de convertir en números complejos el flujo serial de bits que ingresan en éste. El estándar IEEE 802.11n ofrece cuatro opciones de mapeado las que se muestran en la Tabla II.

TABLA II
TIPOS DE MAPEADO DE CONSTELACIÓN EN IEEE 802.11n

	Tipo de Mapeado
1	BPSK
2	QPSK
3	16 - QAM
4	64 - QAM

La ecuación (7) presenta la forma genérica para los datos de salida del bloque Mapeador de Constelación.

$$d = (I + j Q) K_{MOD} \quad (7)$$

donde: I : valor en fase
 Q : valor en cuadratura
 K_{MOD} : factor de normalización

El factor de normalización K_{MOD} tiene como finalidad que se consiga la misma potencia promedio para todos los valores del mapeado.

Finalmente, para el proceso de conversión se debe utilizar el mapeado de constelación mediante el código Gray, cuya principal característica consiste en que dos valores sucesivos difieren únicamente en uno de sus dígitos.

G. Bloque Continuo

El bloque Continuo es el encargado de administrar el funcionamiento de los bloques FIFO Mapeador ubicados en cada uno de los flujos espaciales. Este bloque no se encuentra definido en el estándar IEEE 802.11n, pero es requerido para los casos en los cuales los flujos espaciales utilizan diferentes tipos de mapeado de constelación.

H. Bloque FIFO-Mapeador

El bloque FIFO-Mapeador es utilizado para almacenar los números complejos que se obtienen en las salidas del bloque Mapeador de Constelación con el objetivo de permitir una lectura simultánea de los números complejos en cada uno de los flujos espaciales.

I. Bloque Desplazamiento Cíclico [1], [2]

El bloque Desplazamiento Cíclico es utilizado para prevenir la formación de haces involuntario debido a la transmisión de señales similares a través de los diferentes flujos espaciales. Los bloques de Desplazamiento Cíclico se colocan en los flujos espaciales dos, tres y cuatro. En la Tabla III se presentan los valores de retraso que se introducen en cada flujo espacial.

TABLA III
VALORES DE DESPLAZAMIENTO CÍCLICO PARA EL CAMPO DE DATOS EN IEEE 802.11n [1]

Flujo Espacial	Valor del Desplazamiento Cíclico (ns)
SS1	-
SS2	-400
SS3	-200
SS4	-600

J. Bloque Mapeador Espacial Directo [1]

El bloque Mapeador Espacial Directo es el encargado de enrutar los datos que provienen de los bloques Desplazamiento Cíclico con las respectivas entradas de los bloques Integradores ubicados en cada uno de los flujos espaciales.

K. Bloque Integrador [1], [2]

El bloque Integrador es el encargado de realizar la inserción de los tonos Piloto en las subportadoras correspondientes. El estándar IEEE 802.11n en modo de

operación Greenfield utiliza un canal de 40 MHz que ubica seis subportadoras Piloto en las posiciones -53, -25, -11, 11, 25, 53. Además, el bloque Integrador se encarga de colocar las componentes DC iguales a cero en las subportadoras apropiadas.

La Fig. 3 presenta la ubicación de las subportadoras Piloto, DC y de Datos para un canal de 40 MHz.

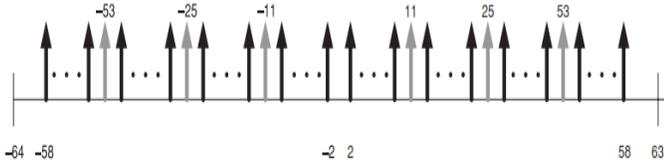


Fig. 3 Ubicación de las subportadoras para un canal de 40 MHz [2]

Para determinar el valor de las subportadoras Piloto que se deben utilizar, el estándar IEEE 802.11n presenta la Tabla IV que resume las diferentes opciones posibles.

TABLA IV
VALORES DE SUBPORTADORAS PILOTO PARA
UN CANAL DE 40 MHz [1]

N _{ss}	i _{ss}	Ψ _{iss,0}	Ψ _{iss,1}	Ψ _{iss,2}	Ψ _{iss,3}	Ψ _{iss,4}	Ψ _{iss,5}
1	1	1	1	1	-1	-1	1
2	1	1	1	-1	-1	-1	-1
2	2	1	1	1	-1	1	1
3	1	1	1	-1	-1	-1	-1
3	2	1	1	1	-1	1	1
3	3	1	-1	1	-1	-1	1
4	1	1	1	-1	-1	-1	-1
4	2	1	1	1	-1	1	1
4	3	1	-1	1	-1	-1	1
4	4	-1	1	1	1	-1	1

Otra función que cumple el bloque Integrador es organizar el proceso de lectura de las subportadoras previo al ingreso al bloque Transformada Inversa de Fourier. Y, finalmente, el bloque Integrador se encarga de realizar una rotación positiva de 90° a las componentes superiores antes de su ingreso al bloque Transformada Inversa de Fourier.

L. Bloque Transformada Inversa de Fourier [1], [4]

El bloque Transformada Inversa de Fourier es utilizado para realizar la modulación OFDM debido a que el algoritmo de la transformada inversa de Fourier discreta se asemeja a la sumatoria obtenida al formar el símbolo OFDM.

La ecuación (8) presenta el algoritmo de la transformada inversa de Fourier discreta; mientras tanto, en (9) se puede apreciar la sumatoria que constituye el símbolo OFDM.

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j \frac{k2\pi n}{N}} \quad n = 0, \dots, N-1 \quad (8)$$

$$s[n] = \text{Re} \left[\sum_{i=0}^{N_c-1} d_i e^{j \frac{i2\pi n}{N_c}} \right] \quad n = 0, \dots, N_c-1 \quad (9)$$

Como se puede evaluar en las ecuaciones presentadas, las expresiones son similares a diferencia del factor 1/N que incluye el algoritmo de la IDFT.

Para aplicaciones prácticas se aplica la transformada rápida de Fourier en lugar de la transformada discreta de Fourier por el hecho que reduce el número de procesos que se realizan en su interior, por lo tanto, es una técnica más eficiente.

M. Bloque Prefijo Cíclico [5]

El bloque Prefijo Cíclico es utilizado con el objetivo de crear símbolos OFDM periódicos, que permitan utilizar la propiedad de convolución circular de la transformada discreta de Fourier, para de esta manera, recuperar en el receptor los datos enviados sin los efectos del canal.

Esta propiedad indica que la aplicación de la convolución circular a señales periódicas en el dominio de tiempo, es similar a la multiplicación de dichas señales en el dominio de la frecuencia. La ecuación (10) resume la propiedad que se acaba de enunciar.

$$x[n] \otimes h[n] \xrightarrow{DFT} X[k]H[k] \quad (10)$$

El bloque Prefijo Cíclico se encarga de copiar la parte final del símbolo OFDM y la coloca al inicio del mismo. La porción del símbolo OFDM que debe ser copiada debe ser al menos igual al ensanchamiento del retardo del canal. Adicionalmente, con el uso del prefijo cíclico se evitan problemas de interferencia intersímbolo e interferencia interportadora, pero esto se produce como resultado de copiar los datos extras al inicio del símbolo OFDM.

El estándar IEEE 802.11n propone dos alternativas de tamaño para el prefijo cíclico. Para la primera alternativa, considera un prefijo cíclico normal cuya duración es de 0.8μs, es decir, se copian los 32 últimos datos del símbolo OFDM y se los coloca al inicio. En conclusión, el símbolo OFDM final tendrá una duración de 4μs y se encuentra formado por un total de 160 valores. En cambio la segunda alternativa es una innovación del estándar IEEE 802.11n que considera la utilización de un prefijo cíclico corto cuya duración es de 0.4μs, es decir, se copian los últimos 16 datos del símbolo OFDM; entonces, el símbolo OFDM resultante tendrá una duración de 3.6μs y está compuesto de 144 valores.

N. Bloque Menú 11n

El bloque Menú 11n tiene por objetivo facilitar la administración de los parámetros encargados de configurar el funcionamiento de cada uno de los bloques que integran el transmisor implementado. Su funcionamiento consiste en determinar los valores de salida de acuerdo al valor MCS ingresado.

Las salidas del bloque Menú 11n son las siguientes:

- ✓ Tipo Mapeado
- ✓ Tasa Codificación
- ✓ N_{ss}
- ✓ N_{es}
- ✓ N_{dbps}
- ✓ N_{cbps}
- ✓ N_{bpsc}

Con la información proporcionada por estas salidas se configura el funcionamiento de los bloques que integran el transmisor.

Adicionalmente, el bloque Menú 11n es el encargado de calcular el número de símbolos OFDM que se obtienen en las salidas del transmisor implementado. Para calcular el número de símbolos OFDM se utiliza (11).

$$N_{SYM} = \left\lceil \frac{8 * longitud + 16 + 6 * N_{ES}}{N_{DBPS}} \right\rceil \quad (11)$$

donde: $\lceil X \rceil$ corresponde al cálculo de la función ceil o techo.
longitud corresponde al tamaño del campo de Datos.

III. IMPLEMENTACIÓN DE LOS BLOQUES QUE INTEGRAN EL TRANSMISOR

Esta sección está dedicada a la implementación de los bloques principales que integran el transmisor. Previo al comienzo de la descripción del proceso de implementación de los bloques se creyó conveniente presentar los requisitos de diseño.

A. Requisitos de Diseño [1], [2]

Para el caso particular del modo de operación Greenfield es necesario configurar algunos parámetros que servirán de base para la implementación de los bloques. Estos parámetros son los siguientes:

1) *Reloj del Transmisor*: El reloj del transmisor tiene una frecuencia de 80 MHz, que es utilizada para controlar el funcionamiento de los bloques sincrónicos que forman el transmisor.

2) *Frecuencia de Trabajo*: La frecuencia de trabajo que utilizan los bloques depende del funcionamiento interno de los mismos. Es así, que se determinó una frecuencia de trabajo de 40 MHz para los bloques comprendidos entre el bloque Aleatorizador y los bloques Codificador Convolutacional Binario; para los bloques restantes se utiliza una frecuencia de trabajo de 80 MHz.

3) *Ancho de Banda*: El modo de operación Greenfield utiliza un canal de 40MHz.

4) *Esquema de Codificación y Modulación*: El estándar IEEE 802.11n permite utilizar setenta y cinco opciones de codificación y modulación. Cada una de las alternativas de MCS (*Modulation Coding Scheme*) ofrece variaciones en: número de cadenas de transmisión, tipo de mapeado, tasa de codificación y números de codificadores convolutacionales.

5) *Flujos Espaciales*: El transmisor puede habilitar desde uno hasta cuatro flujos espaciales.

6) *Número de Codificadores BCC*: Se permite el uso de uno o dos codificadores BCCs (*Binary Convolutional Code*).

7) *Tasas de Codificación*: Se presentan cuatro alternativas de tasas de codificación como se mencionó en la sección II.C.

8) *Tipos de Mapeado de Constelación*: Se permiten cuatro tipos de mapeado de constelación como se indicó en la sección II.F.

9) *Tipos de Prefijo Cíclico*: El estándar IEEE 802.11n permite el uso de dos tipos de prefijos cíclico: corto y normal.

10) *Subportadoras Piloto, DC y Datos*: Se han definido ciento ocho subportadoras de datos, seis subportadoras Piloto ubicadas en las posiciones -53, -25, -11, 11, 25, 53 y catorce subportadoras DC.

B. Implementación de los bloques

A continuación, se describe el proceso de implementación de cada uno de los bloques que integran la sección de banda base del transmisor implementado.

1) *Bloque Aleatorizador*: La implementación del bloque Aleatorizador se llevó a cabo mediante la implementación de tres procesos. En la Fig. 4 se pueden apreciar los procesos.



Fig. 4 Procesos del bloque Aleatorizador

2) *Bloque Analizador de Codificación*: Para la implementación fue necesario definir un proceso que se encargue de distribuir los bits Fig. 5.



Fig. 5 Proceso del bloque Analizador de Codificación

3) *Bloque Codificador Convolutacional Binario*: La implementación del bloque Codificador Convolutacional Binario se consiguió mediante un desarrollo modular de los procesos que se presentan en la Fig. 6.

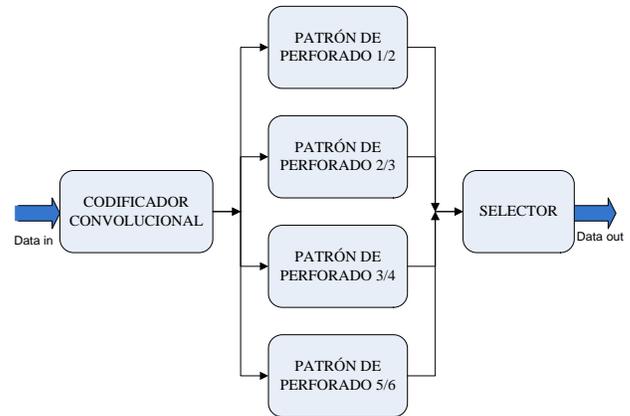


Fig. 6 Procesos internos del bloque Codificador Convolutacional Binario

4) *Bloque Analizador de Flujos*: La implementación del bloque Analizador de Flujos requiere duplicar la frecuencia de trabajo con respecto a los bloques previos para evitar la pérdida de bits; por tal motivo, se utilizó un bloque que dupliquen la frecuencia de muestreo. Adicionalmente, se implementaron otros procesos para obtener el funcionamiento deseado del bloque Analizador de Flujos. En la Fig. 7 se presentan los procesos internos.

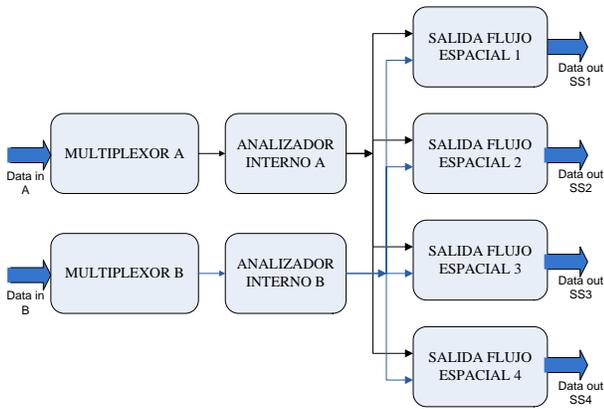


Fig. 7 Procesos internos del bloque Analizador de Flujos

5) *Bloque Intercalador:* El bloque Intercalador requirió un análisis muy particular para su implementación debido a que los grupos que deben ser intercalados tienen diferentes tamaños, por lo tanto, el cálculo de los coeficientes varía.

Para la implementación del bloque Intercalador se determinaron dos procesos, los cuales se presentan en la Fig. 8.

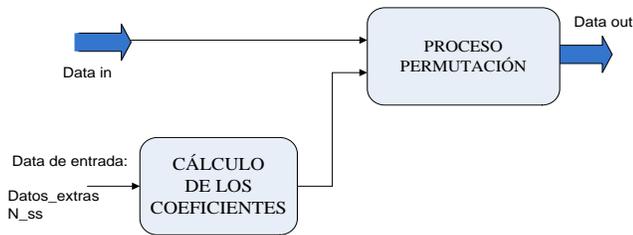


Fig. 8 Procesos internos del bloque Intercalador

El primer proceso corresponde al Cálculo de los Coeficientes cuya función es determinar el valor de los coeficientes de las permutaciones presentadas en la sección II.E. Para el cálculo de los coeficientes se utilizan las ecuaciones presentadas en el estándar. El segundo proceso corresponde en sí al proceso de permutación. Se diseñó un bloque denominado bloque Lectura-Escritura que permite realizar el proceso de lectura los bits almacenados previamente; mientras se almacenan los bits que ingresan al bloque, con la implementación de este bloque se pueden realizar estos procesos de manera simultánea. Debido a la flexibilidad que presenta el bloque de Lectura-Escritura puede ser reutilizado en otros bloques que requieren un funcionamiento similar, por ejemplo, bloques como: Integrador y Prefijo Cíclico.

La Fig. 9 presenta los bloques que integran el bloque Cálculo de los Coeficientes. Mientras tanto, en la Fig. 10 se presentan los bloques internos del bloque Proceso Permutación.

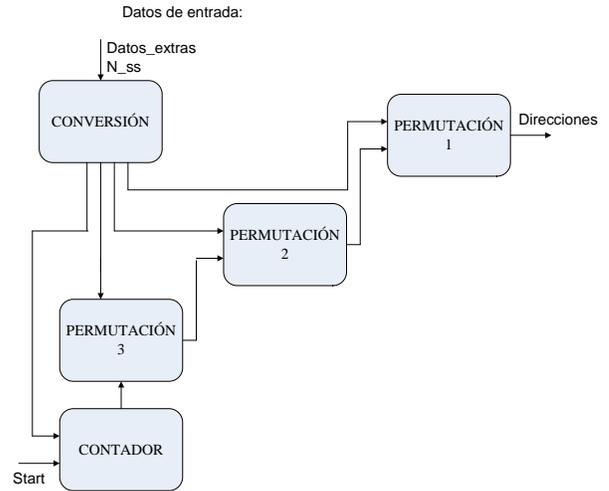


Fig. 9 Bloques internos del proceso Cálculo de Coeficientes

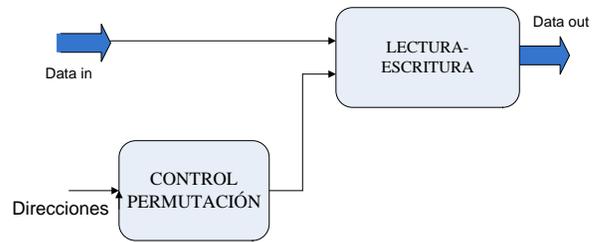


Fig. 10 Bloques internos del bloque Proceso Permutación

Tomando como referencia la Fig. 1 se debe determinar el conjunto de bloques implementados en cada uno de los flujos espaciales.

6) *Bloque Mapeador de Constelación:* Para la implementación se definieron seis procesos que permiten un adecuado funcionamiento del bloque. Estos procesos se pueden apreciar en la Fig. 11.

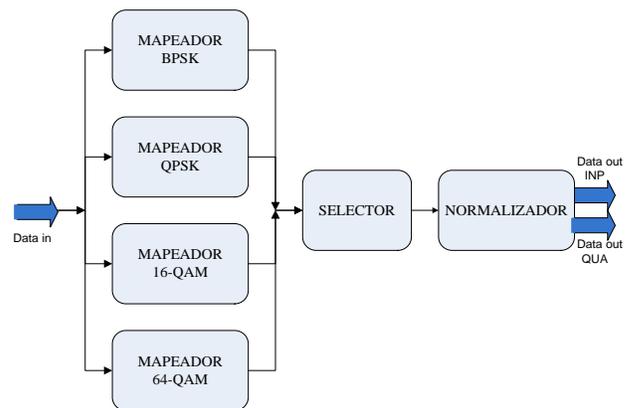


Fig. 11 Procesos internos del bloque Mapeador de Constelación

7) *Bloque Continuo*: Para la implementación de este bloque se consideraron algunos procesos que se deben realizar en su interior. La Fig. 12 muestra los procesos internos del bloque Continuo.

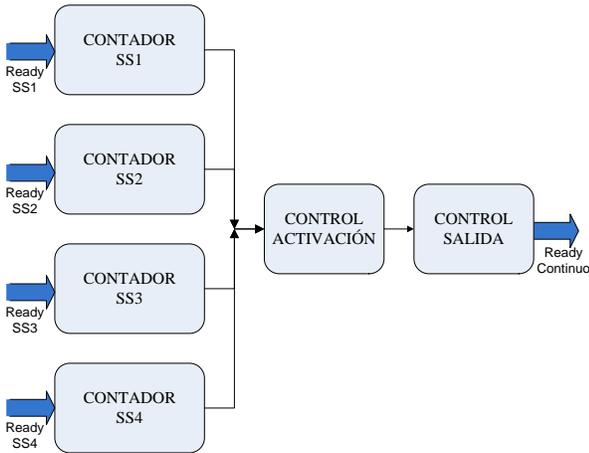


Fig. 12 Procesos internos del bloque Continuo

8) *Bloque FIFO Mapeador*: Para la implementación de este bloque se necesitó utilizar dos memorias FIFO [4], una para almacenar los datos “en fase” y la otra para los datos “en cuadratura” como se aprecia en la Fig. 13. El control de este bloque lo realiza el bloque Continuo.

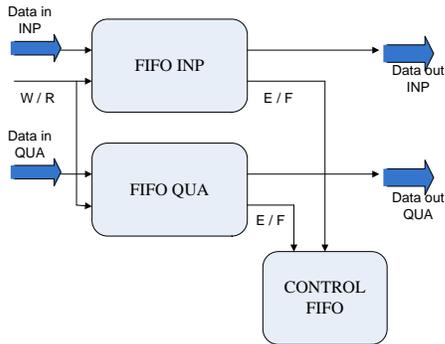


Fig. 13 Procesos internos del bloque FIFO Mapeador

La entrada del bloque correspondiente a W/R representa los puertos de escritura (W, *write*) y lectura (R, *read*). Mientras tanto, el puerto E/F es un puerto de control que informa cuando la memoria se encuentra vacía (E, *empty*) y/o llena (F, *full*).

9) *Bloque Desplazamiento Cíclico (CSD)*: La implementación del bloque CSD es similar al bloque FIFO Mapeador, la única diferencia consiste que el control de las memorias FIFO es realizado por el bloque CSD. La Fig. 14 presenta la estructura del bloque CSD.

10) *Bloque Mapeador Espacial Directo*: La implementación de este bloque es sencilla debido a que únicamente realiza la asignación de las señales de entrada a las salidas correspondientes. En la Fig. 15 se puede apreciar el bloque Mapeador Espacial Directo.

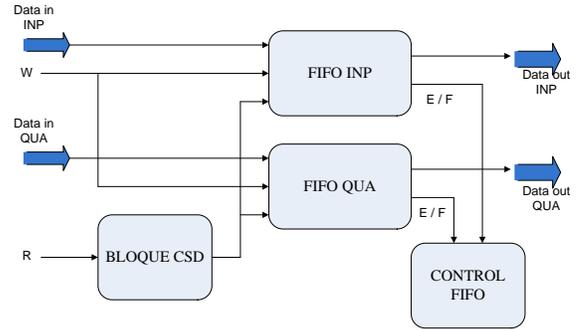


Fig. 14 Procesos internos del bloque CSD

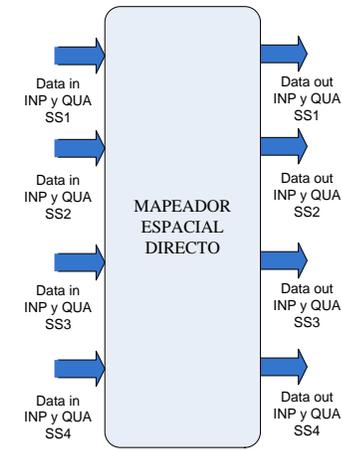


Fig. 15 Bloque Mapeador Espacial Directo

11) *Bloque Integrador*: La implementación de este bloque consistió básicamente en tres procesos los cuales consisten en: realizar el direccionamiento para almacenar los datos que ingresan, inserción de las subportadoras Piloto y, finalmente, la rotación de 90° en sentido positivo de las componentes superiores utilizadas para formar el símbolo OFDM.

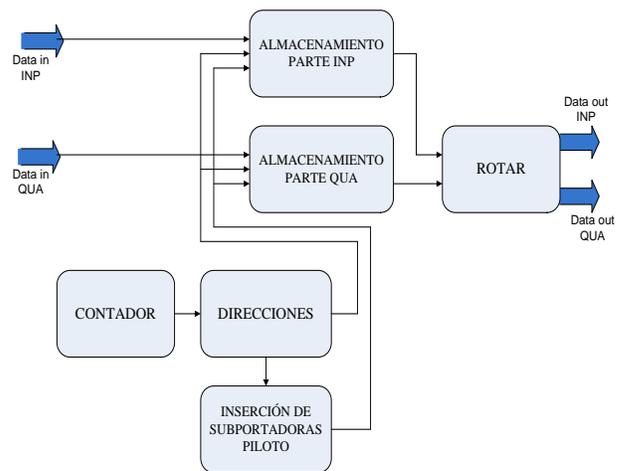


Fig. 16 Bloques internos del bloque Integrador

En la Fig. 16 se presentan los bloques principales que conforman el bloque Integrador. El bloque denominado

Almacenamiento está implementado a partir del bloque Lectura-Escritura definido en la sección III.B.5.

Adicionalmente, es importante indicar la ubicación de las subportadoras Piloto, DC y de Datos previo a su ingreso en el bloque Transformada Inversa de Fourier. La ubicación de las subportadoras se presenta en la Fig. 17.

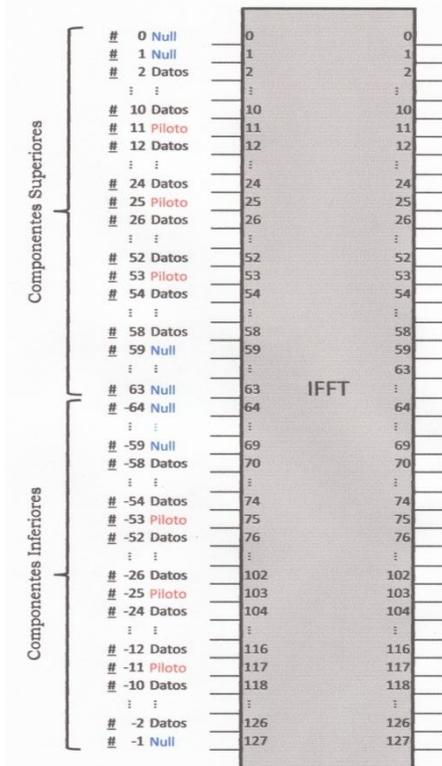


Fig. 17 Organización de las subportadoras Piloto, DC y Datos previo al ingreso al bloque Transformada Inversa de Fourier

12) *Bloque Transformada Inversa de Fourier:* Para la implementación de la IFFT se utilizó el bloque “Fast Fourier Transform 6.0” que forma parte de la librería DSP perteneciente al toolbox Xilinx Blockset de System Generator.

La Fig. 18 presenta los bloques utilizados para el funcionamiento de la Transformada Inversa de Fourier.

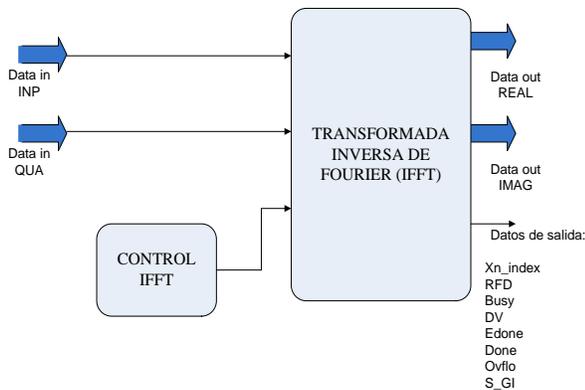


Fig. 18 Bloque Transformada Inversa de Fourier

La configuración requerida del bloque IFFT para el presente proyecto se presenta en la Tabla V.

TABLA V
CONFIGURACIÓN DEL BLOQUE IFFT PARA UN CANAL DE 40 MHz [3]

PARÁMETROS	VALOR
Arquitectura	Pipelined Streaming I/O
Longitud	128 puntos
Opción de escala	Escalable
Modo de redondeo	Truncado
Orden de salida	Orden natural
Memorias	Bloque RAM
FWD_INV	'0' (IFFT)
SCALE_SCH	"01101111"

Los puertos de control del bloque IFFT facilitan la implementación del bloque Prefijo Cíclico, estos valores corresponden a: X_n_index indica la referencia posicional de los valores de salida, RFD utilizado para el informar la disponibilidad para el ingreso de datos, $Busy$ informa que el bloque se encuentra procesando los datos ingresados, DV anuncia la salida de los datos válidos, $Edone$ y $Done$ informan la finalización de cálculo de la IFFT, $Ovflto$ utilizado para indicar el desbordamiento de los cálculos.

13) *Bloque Prefijo Cíclico:* Para la implementación de este bloque se tomó en cuenta las dos opciones de prefijo cíclico permitidas por el estándar IEEE 802.11n; por lo tanto, la implementación del bloque consistió en realizar un control para contabilizar el número de valores que deben ser copiados al inicio del símbolo OFDM. La Fig. 19 muestra las ideas básicas para la implementación del bloque Prefijo Cíclico.

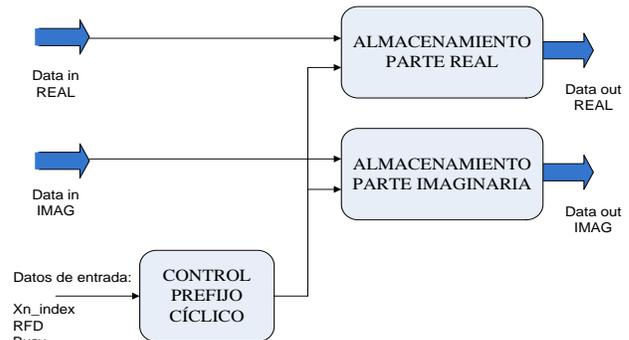


Fig. 19 Bloque Prefijo Cíclico

14) *Bloque Menú 11n:* Para la implementación del bloque Menú 11n se crearon tablas que resuman los valores de las diferentes opciones de MCS que permite el estándar IEEE 802.11n [1]. La Fig. 20 presenta el bloque Menú 11n implementado.

Para el cálculo del número de símbolos OFDM se utiliza (11) presentada en la sección II.N.

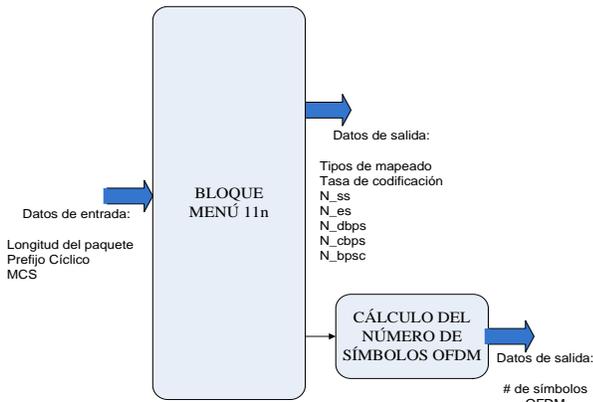


Fig. 20 Bloque Menú 11n

15) *Bloque Transmisor*: Para la implementación del transmisor se integran los bloques que han sido implementados tomando como referencia la Fig. 1. Adicionalmente, es necesario incorporar algunos bloques extras propios de SysGen [5] para permitir acoplar los diferentes bloques.

IV. PRUEBAS Y RESULTADOS OBTENIDOS

Las pruebas realizadas se basaron en comparar que los resultados obtenidos mediante los procesos de hardware de co-simulación [6] sean similares a los conseguidos en las simulaciones. Para realizar las pruebas se utilizó la tarjeta de entrenamiento Virtex 5LX 110T de Xilinx [7], además de las librerías de System Generator incorporadas en el entorno de Simulink. En la Fig. 21 se presentan los equipos utilizados para ejecutar las pruebas.



Fig. 21 Equipos utilizados para el desarrollo de las pruebas

Para cada uno de los bloques, así como para el transmisor completo se realizaron las pruebas descritas obteniendo los resultados esperados. La Fig. 22 (a) muestra un ejemplo de los resultados obtenidos mediante la simulación del bloque Mapeador de Constelación; mientras tanto, la Fig. 22 (b) presenta los resultados del proceso de hardware de co-simulación del bloque Mapeador de Constelación.

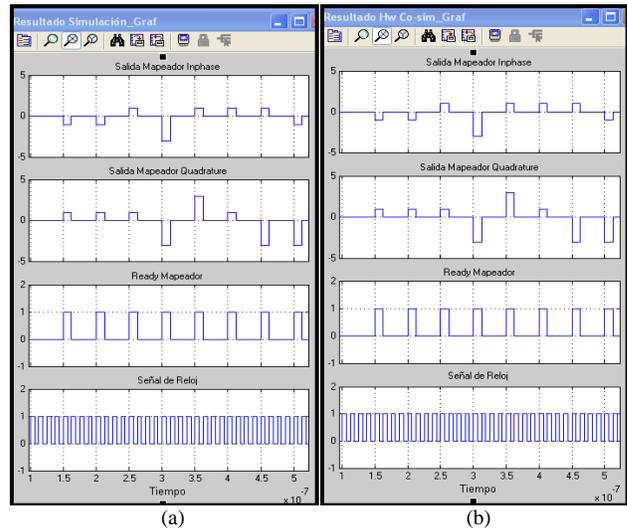


Fig. 22 Resultados de la simulación y del proceso de hardware de co-simulación del bloque Mapeador de Constelación

Entre las pruebas realizadas al transmisor completo se utilizaron las siguientes opciones:

- 1) *Cambiando el Número de Cadenas Transmisoras*: Dependiendo del MCS que se seleccione se determina el número de cadenas que se habilitan. Las pruebas verificaron que el transmisor cumple con habilitar las cadenas de transmisión que indica el MCS seleccionado.
- 2) *Cambiando el Mapeado de Constelación*: Las pruebas comprobaron que se puede tener diferentes tipos de mapeado de constelación en cada uno de los flujos espaciales.
- 3) *Cambiando la tasa de codificación*: La tasa de codificación está determinada por el esquema de codificación y modulación que se seleccione. La tasa de codificación determina el comportamiento y la generación de los datos en el interior del transmisor.

V. CONCLUSIONES

- La empresa Xilinx proporciona una amplia variedad de herramientas útiles para el diseño, verificación, y simulación de los prototipos implementados mediante descripción de hardware. Para el proyecto presentado se utilizaron los programas: ISE Evaluation 11 (*Integrated Software Environment*) y SysGen (*System Generator*).
- Debe existir una adecuada coordinación y sincronización entre los bloques. Particularmente con las señales de control que controlan el funcionamiento de los bloques.
- Una vez realizada la implementación, se procedió a comprobar el funcionamiento de cada uno de los bloques así como del transmisor completo.
- Para la generación del bloque que represente el procedimiento de hardware de co-simulación se requiere que los bloques implementados no tengan errores. El tiempo de generación del bloque depende de la composición del bloque, lo que determina la complejidad para realizar los procesos de síntesis, colocación y enrutado del bloque implementado.

- En la Tabla VI se presenta un resumen de los recursos lógicos usados en la implementación de la sección de banda base del transmisor desarrollado en el presente proyecto. Es importante mencionar que la capacidad de la tarjeta de entrenamiento Virtex 5LX 110T facilitó el proceso de diseño.

TABLA VI
RESUMEN DE RECURSOS LÓGICOS UTILIZADOS EN EL SECCIÓN DE BANDA BASE DEL TRANSMISOR IMPLEMENTADO

RECURSOS USADOS	CANTIDAD	PORCENTAJE DE USO
Número de Registros	18,007 de 69,120	26 %
Número de LUTs	22,839 de 69,120	33 %
Número de Rutas	2,844 de 138,240	2 %
Número de BlockRAM	49 de 148	33 %
Número de BUFG	19 de 32	59 %
Número de BSCANs	1 de 4	25 %
Número de DSP48Es	40 de 64	62 %

- Las pruebas consistieron básicamente en cambiar el valor MCS (selecciona el esquema de codificación y modulación), es así, que dependiendo del valor de MCS se determinan los recursos necesarios para su funcionamiento.

VI. REFERENCIAS

- [1] Wireless LAN Medium Access Control (MAC) & Physical Layer (PHY) Specifications, IEEE 802.11n, 2009.
- [2] E. Perahia y R. Stacey, *Next Generation Wireless LANs Throughput, Robustness, and Reliability in 802.11n*, Cambridge University Press 2008.
- [3] (2011) Sitio web de Xilinx, *Hoja de datos del bloque Transformada Rápida de Fourier*. Disponible en: http://www.xilinx.com/support/documentation/ip_documentation/xfft_d_s260.pdf
- [4] (2011) Sitio web de Xilinx, *Hoja de datos de LogiCORE IP Generador FIFO v5.1*. Disponible en: http://www.mimuw.edu.pl/~marpe/pul/hdl/w7_serialrcv/ipcore_dir/fifo_generator_ug175.pdf
- [5] (2008) Sitio web de Xilinx, *Guía de Referencia de System Generator para DSP*. Disponible en: http://www.xilinx.com/support/sw_manuals/sysgen_ref.pdf
- [6] (2008) Sitio web de Xilinx, *Guía de Usuario de System Generator para DSP*. Disponible en: http://www.xilinx.com/support/sw_manuals/sysgen_user.pdf
- [7] (2011) Sitio web de Xilinx, *Guía de Usuario de la Plataforma de Evaluación ML505/ML506/ML507*. Disponible en: http://www.xilinx.com/support/documentation/boards_and_kits/ug347.pdf

VII. BIOGRAFÍAS



Iván M. Bernal, graduado del Instituto Nacional Mejía. Obtuvo el título de Ingeniero en Electrónica y Telecomunicaciones en la Escuela Politécnica Nacional en 1992. Obtuvo los títulos de M.Sc. (1997) y Ph.D. (2002) en *Computer Engineering* en *Syracuse University*, NY, USA. Ha realizado cursos especializados en varios países europeos, latinoamericanos, Estados Unidos y en Corea del Sur. Actualmente trabaja en la EPN, en el Departamento de Electrónica, Telecomunicaciones y Redes de Información (DETRI).



Romel Iván M. Espinosa Reyes, nació en Quito-Ecuador el 20 de agosto de 1985. Realizó sus estudios secundarios en el Colegio Militar “Trn. Lauro Guerrero” de la ciudad de Loja, donde obtuvo el título de Bachiller en Físico-Matemático. En el año 2003 ingresó a la Escuela Politécnica Nacional, en donde obtuvo el título de Ingeniero en Electrónica y Telecomunicaciones en agosto del 2011. Actualmente, se desempeña como Ingeniero Junior en Radio Frecuencia en SESTEL-KONNECTADO. Áreas de interés: comunicaciones celulares, en particular optimización de redes celulares que utilizan tecnologías UMTS y/o GSM.