

Entrenador de comunicaciones digitales basado en FPGA controlado mediante LabVIEW

Oswaldo González, Manuel Rodríguez, Alejandro Ayala, Silvestre Rodríguez, Beatriz R. Mendoza
 Dpto. Ingeniería Industrial, Escuela Superior de Ingeniería y Tecnología, Universidad de La Laguna
 Avda. Astrofísico Francisco Sánchez, S/N, 38203 La Laguna, Tenerife, Islas Canarias
 Email: oghdez@ull.es, mrvalido@ull.es, aayala@ull.es, srdguezp@ull.es, bmendoza@ull.es

Abstract—This paper describes the design and implementation on FPGA (Field Programmable Gate Array) of a training system for learning the basics about digital communications. The developed system is able to generate the main digital modulated signals (ASK, FSK, BPSK, M-PSK, M-QAM) and it also contains an optimum receiver for demodulating these signals when they are corrupted by AWGN (Additive White Gaussian Noise). An AWGN generator is included into the same FPGA in case an external one is not available. Moreover, a software tool developed in LabVIEW is provided to analyze the different signals at the modulator and the demodulator. This work constitutes a tool for helping students to improve their understanding and learning about digital communication. Nevertheless, its design and implementation could also be focused as a Final Graduate Project for a bachelor's degree in Electronics Engineering. The project covers several topics on communications, signal processing, digital-to-analog conversion and programming in LabVIEW, as well as design and synthesis over logic devices by using hardware description languages. Therefore, it allows the student to go into the previous topics in a greater depth and it prepares him/her for applying the knowledge of them in the achievement of a specific objective. On the other hand, the project constitutes the solution of a non-trivial problem by using digital techniques, then making the student more confident to cope with new challenges in the future. From an educational point of view, this tool has been used in digital communication courses to assess its ability to improve students' learning. It has been demonstrated an above 30 % enhance on academic students' marks in digital communication topics since the introduction of practical sessions with this tool as part of subject program.

I. INTRODUCCIÓN

Los entrenadores de comunicaciones configuran una herramienta muy eficiente para facilitar a los alumnos la comprensión de los aspectos relacionados con los sistemas de comunicación [1]–[3], al permitirles trabajar con dispositivos que generan y procesan señales reales y no meramente con modelos matemáticos e ilustraciones expuestas en una pizarra o diapositiva. Si estos módulos *hardware* cuentan además con algún programa *software* que les permita analizar gráficamente qué está pasando en distintos puntos del entrenador, es posible acelerar aún más dicho aprendizaje.

En los tiempos actuales, no contamos con los recursos económicos necesarios para adquirir este tipo de productos y que además se adapten perfectamente a nuestras necesidades. Es por ello que en este trabajo se plantea el diseño e implementación de un entrenador de comunicaciones digitales de bajo coste y lo suficientemente completo para abordar el

aprendizaje de muchos de los conceptos básicos sobre este tipo de sistemas de comunicación.

El diseño se ha decidido que esté basado en una FPGA, dadas las posibilidades de este tipo de productos para desarrollar sistemas de procesamiento digital eficientes y de bajo coste, además que se han demostrado muy adecuadas para este tipo de aplicaciones [2], [4]–[7]. Aunque se plantea este trabajo como un producto ya desarrollado para el aprendizaje de los esquemas de comunicación digital, es posible también enfocarlo como un proyecto de fin de carrera donde el objetivo sea el diseño del mismo propiamente dicho.

De esta forma, tras una breve descripción general del sistema en la sección II, la sección III detalla los diferentes elementos a tener en cuenta para llevar a cabo la implementación del entrenador. La sección IV presenta los resultados obtenidos tras el uso del mismo como parte del programa de prácticas de una asignatura sobre ‘Comunicaciones Digitales’ en relación al rendimiento académico del alumnado previo a su implantación. Por último, se finaliza con la exposición de las principales conclusiones de este trabajo en la sección V.

II. DESCRIPCIÓN GENERAL DEL SISTEMA

En la Fig. 1 se muestra el diagrama de bloques del entrenador a implementar. Como se puede observar, la mayor parte de las tareas a ejecutarse (generación de datos, modulación de los mismos, generación del ruido y demodulación de la señal recibida) recaen sobre la FPGA, que incluso controla la transferencia de información hacia el exterior mediante un conversor digital-analógico (DAC, *Digital-to-Analog Converter*) y un módulo UART (*Universal Asynchronous Receiver Transmitter*).

Adicionalmente, se requiere de un *software* instalado en el ordenador personal para poder analizar en tiempo real las distintas señales generadas en la FPGA, cuyos valores se reciben a través de su puerto serie.

III. DISEÑO DEL ENTRENADOR

Tal como se ha comentado previamente, en el desarrollo de este proyecto se optó por una realización digital del entrenador mediante el uso de lenguajes de descripción de *hardware* de alto nivel, en concreto VHDL (*Very high speed integrated circuit – Hardware Description Language*), para su posterior implementación en una FPGA comercial y prueba del mismo.

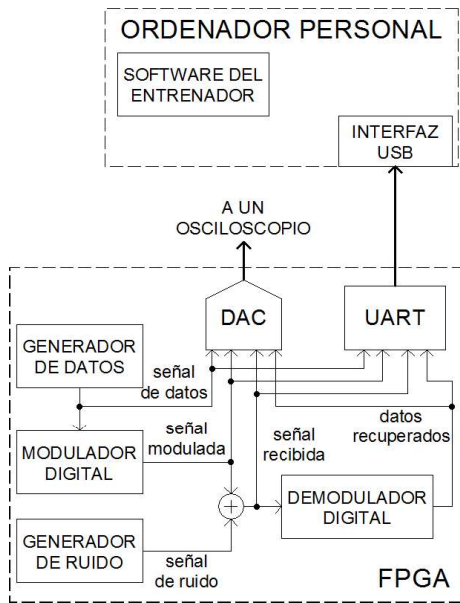


Fig. 1. Esquema de bloques del entrenador

En los siguientes apartados se describirán los aspectos básicos necesarios para poder llevar a cabo la implementación del entrenador en una FPGA. En particular, internamente se trabajará con palabras de datos por lo que es importante definir el formato a utilizar. Por otro lado, la forma de generar las señales portadoras requeridas por el modulador y el demodulador son igualmente de especial relevancia.

III-A. Formato de los datos

Se utilizó notación de punto fijo para definir el formato utilizado por los datos procesados por las diferentes estructuras del diseño realizado. En dicho formato de datos, de los N bits que constituye cada palabra de datos, se consideró notación en complemento a dos, de tal forma que el bit más significativo indica el signo, mientras que un número M de los bits menos significativos (LSB, *Least Significant Bit*) se utilizan para la parte decimal (véase la Fig. 2).

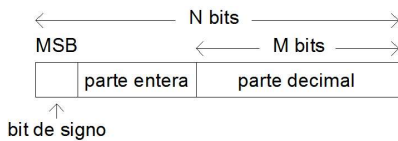


Fig. 2. Formato de la palabra de datos

Las FPGA comerciales suelen disponer de circuitos multiplicadores implementados internamente, por lo que los procesos de multiplicación pueden definirse fácilmente en VHDL y tienen una síntesis directa en la FPGA. Sin embargo, en este caso debe tenerse en cuenta que al multiplicar dos palabras tal como se definen en la Fig. 2, la palabra resultante será de $2N$ bits con una parte decimal de $2M$ bits (ver Fig. 3). Siempre que se trabaje con operandos que no produzcan *overflow*

durante el proceso de multiplicación, esto es, que el resultado obtenido sea representable con N bits, es posible extraer, de la palabra resultante de $2N$ bits, aquéllos comprendidos entre la posición M y la posición $N + M - 1$ (considerando como LSB al bit 0) para configurar este resultado de N bits.

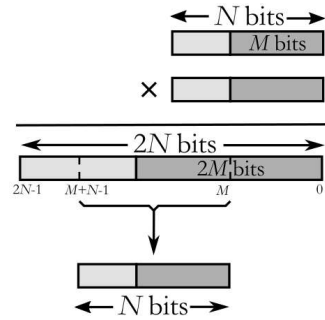


Fig. 3. Proceso de multiplicación de dos palabras de datos

III-B. Generación de las portadoras

La generación de las ondas portadoras (en el modulador y demodulador) se basa en una tabla de K posiciones que contiene las muestras de una forma de onda senoidal y dos punteros de dirección a la misma, uno de ellos desfasado 90° respecto del otro, o lo que es lo mismo, $K/4$ posiciones (véase la Fig. 4). Cada uno de los datos de la tabla utiliza el formato descrito en la Fig. 2.

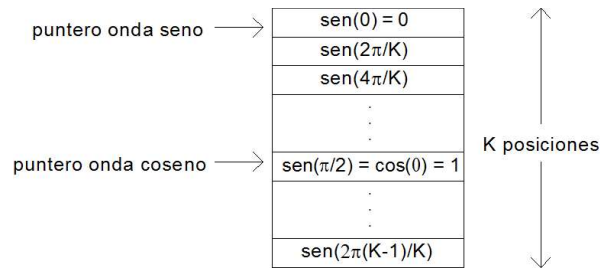


Fig. 4. Tabla de datos que describe las formas de onda senoidales

A cada flanco de la señal de reloj que controla el generador de portadora, los punteros incrementarán en una unidad su posición, describiendo la forma de la onda seno (y coseno). La portadora desfasada 90° (coseno) sólo será necesaria para modulaciones con componentes en cuadratura (QPSK, M-PSK o M-QAM) o por el demodulador. Para modulaciones FSK basta con aplicar desplazamientos a los punteros en cada ciclo de reloj diferentes a la unidad, obteniéndose ondas de frecuencias múltiplos de la de la señal de referencia definida por la tabla.

III-C. Generador de datos

El generador de datos se basa en el uso de registros de desplazamiento realimentados (FSR, *Feedback Shift Register*), que ofrecen secuencias pseudoaleatorias de máxima longitud ($P = 2^R - 1$ bits, donde R es el número de biestables) para

determinadas combinaciones de realimentación a través de una puerta XOR. En la Fig. 5 se muestra la estructura para el generador de $R = 5$ biestables implementado en nuestro caso, donde se requiere realimentar las salidas del segundo y último para obtener una secuencia de longitud máxima.

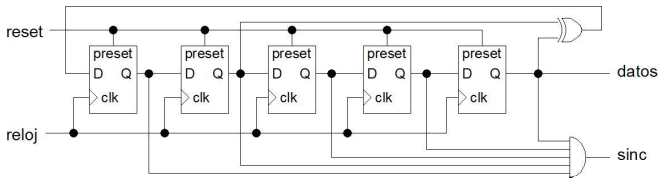


Fig. 5. Estructura del generador de datos

Los biestables deben inicializarse a nivel lógico '1', pues un valor '0' en todos ellos provoca que el FSR presente de manera indefinida un nivel lógico bajo a su salida. Cuando están adecuadamente realimentados, el contenido de los biestables del generador llega a conformar, de manera pseudoaleatoria y con igual probabilidad, todas las palabras binarias posibles con R bits salvo "00...0". Así, se han tomado los n LSB del generador de la Fig. 5 para disponer del dato a utilizar en cada caso por el modulador. Por ejemplo, para modulación 16-QAM se tomaban los $n = 4$ LSB, mientras que para QPSK sólo se requería de los $n = 2$ LSB.

III-D. Generador de ruido

Para obtener la señal de ruido AWGN se creó una tabla que contenía 256 valores que configuraban la función de distribución de probabilidad de un proceso aleatorio gaussiano. Luego, con un generador de datos como el de la Fig. 5 (en este caso, de 21 registros) se obtenía una palabra pseudoaleatoria de 8 bits. La palabra obtenida se iba comparando con los distintos valores de la tabla con la función de distribución gaussiana, hasta encontrar aquél más próximo. La posición en la tabla de dicho valor se hacía corresponder con la amplitud del ruido aleatorio para la muestra actual, a falta de aplicar algún factor de ganancia seleccionable por el usuario. De esta forma, se obtendrán más muestras de ruido con amplitudes próximas a cero, donde la función de distribución gaussiana experimenta mayores variaciones de su valor, dado que la búsqueda se realiza a través de un valor pseudoaleatorio con

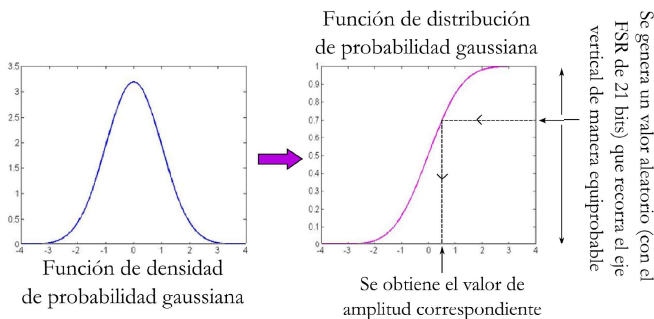


Fig. 6. Generación de las muestras de ruido

una distribución uniforme. La Fig. 6 esquematiza el proceso de obtención de las amplitudes para las muestras de ruido.

En la Fig. 7 se muestra cómo el generador de ruido implementado proporciona una distribución de la amplitud de las muestras de ruido con una función de densidad de probabilidad gaussiana, demostrando un correcto funcionamiento.

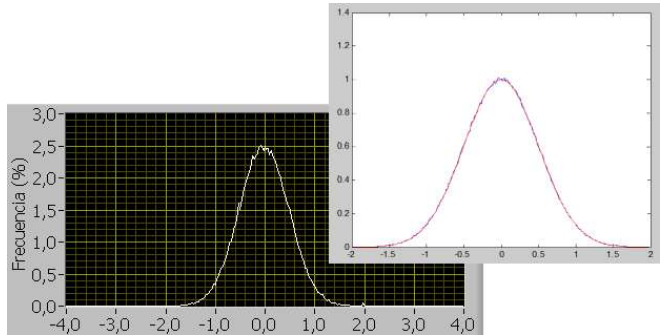


Fig. 7. Distribución de las muestras de ruido (amplitudes) durante una ejecución

III-E. Modulador digital

Para el diseño del modulador se optó por una estructura genérica donde dos portadoras en cuadratura son moduladas en amplitud por las componentes en fase x_i y en cuadratura y_i del símbolo i -ésimo a transmitir, obtenidos de la constelación correspondiente a partir de la palabra pseudoaleatoria de n bits suministrada por el generador de datos:

$$s_i(t) = x_i \sin \omega_c t + y_i \cos \omega_c t, \quad iT < t < (i + 1)T \quad (1)$$

donde $f_c = \omega_c/2\pi$ es la frecuencia de la señal portadora definida por el número K de muestras de la tabla de la Fig. 4 y la frecuencia del reloj del puntero. T es el tiempo de símbolo. Para señales binarias, donde $n = 1$ (BPSK, FSK y ASK), los símbolos no tienen componente en cuadratura ($y_i = 0$).

III-F. Demodulador de datos

Para señales corrompidas por ruido AWGN, el detector óptimo consiste en *filtros de correlación* sincronizados a las portadoras en fase y en cuadratura [8]. A la salida de los correladores, el detector debe determinar el símbolo de la constelación que más se aproxime al símbolo demodulado.

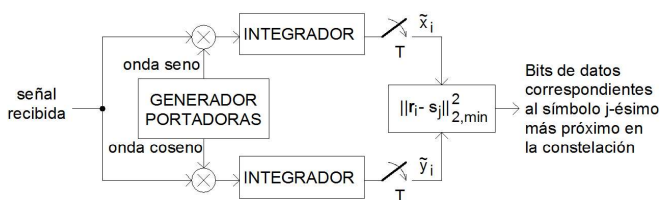


Fig. 8. Estructura del demodulador

En la Fig. 8 se muestra la estructura del demodulador implementado. Se ha supuesto que se dispone de una sincronización

perfecta entre el generador de portadoras del demodulador y la señal recibida, aunque el entrenador permite la introducción de retardos para observar el efecto que producen sobre la demodulación de los datos. Para constelaciones complejas como 16-QAM, el símbolo recibido $\mathbf{r}_i = \tilde{x}_i + j\tilde{y}_i$ debe compararse con cada uno de los $L = 2^n$ símbolos de la constelación \mathcal{M}^L y determinar aquél que minimice la distancia euclídea entre ambos:

$$\check{s}_j = \arg \min_{s_j \in \mathcal{M}^L} \|\mathbf{r}_i - s_j\|_2^2 = \arg \min_{s_j \in \mathcal{M}^L} [(\tilde{x}_i - x_j)^2 + (\tilde{y}_i - y_j)^2] \quad (2)$$

donde $\check{s}_j = \check{x}_j + j\check{y}_j$ es el símbolo más probable de haber sido transmitido y $\|\cdot\|_2^2$ denota la norma euclídea al cuadrado. Para constelaciones más simples como QPSK o modulaciones binarias es posible sustituir el bloque de búsqueda del símbolo más próximo por comparadores donde el umbral se establece al punto medio entre los posibles símbolos recibidos.

IV. RESULTADOS

La Fig. 9 muestra una instantánea del entrenador implementado, donde las señales suministradas por la tarjeta que contiene la FPGA son representadas en un osciloscopio, así como en el *software* residente en el ordenador que se ha desarrollado utilizando LabVIEW®. La FPGA por la que se ha optado finalmente es una Spartan 3A [9], de la que apenas se ha requerido un 25% de sus recursos lógicos para la implementación del entrenador.

Como se ha comentado, el entrenador permite la visualización de las principales señales generadas en el mismo a través de la observación de los cuatro canales de salida de su convertor D/A. La Fig. 10 muestra la adquisición con un osciloscopio de dichas señales para el caso de modulación FSK (señal transmitida, datos transmitidos, señal recibida con ruido y datos demodulados). Podemos observar un perfecto funcionamiento del demodulador, donde los datos demodulados coinciden con los transmitidos con un retardo de un bit, correspondiente al tiempo de símbolo T (el requerido por los

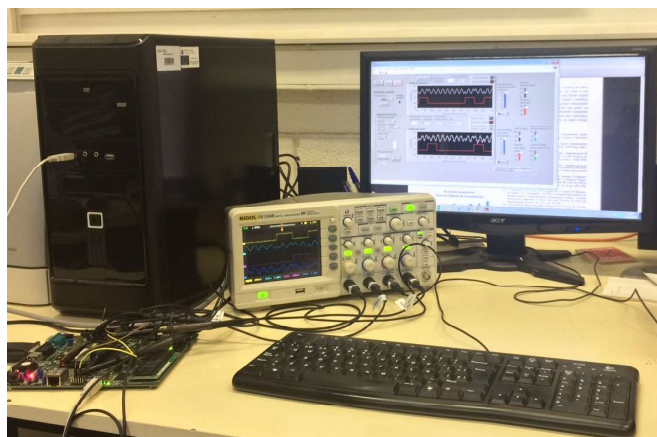


Fig. 9. Visual del entrenador

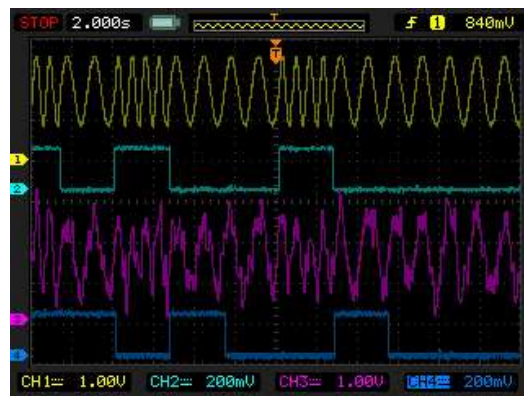


Fig. 10. Visualización de las señales del entrenador en un osciloscopio

integradores de la Fig. 8 para obtener la estimación de los símbolos recibidos).

En relación al *software* desarrollado para analizar las señales suministradas por el entrenador de comunicaciones digitales, la Fig. 11 muestra dos instantáneas de los paneles principal y secundario que componen la interfaz de usuario.

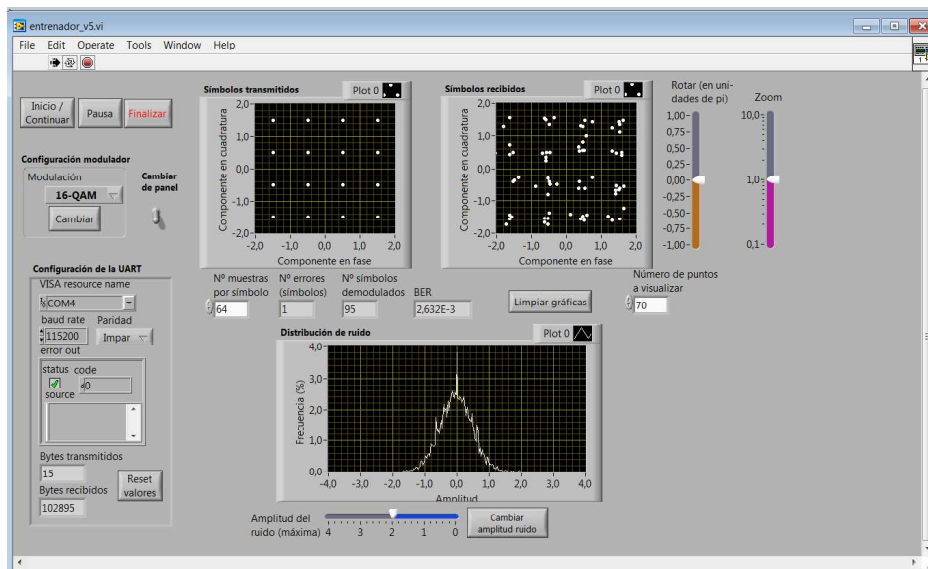
El panel principal muestra la evolución temporal de las señales tanto en el modulador como en el demodulador, en este caso para modulación QPSK, donde además se proporciona información sobre el número de errores cometidos durante la demodulación de los datos si los hubiese habido. Obsérvese que los datos demodulados (señal en rojo en la gráfica inferior del panel) están retrasados un tiempo de símbolo T respecto de los datos originalmente transmitidos (señal en rojo en la gráfica superior), por lo que se comprueba que en esta captura no se han producido errores de transmisión.

El segundo panel puede utilizarse para analizar las constelaciones de las señales transmitidas y recibidas, así como la distribución de las muestras de ruido. La figura presenta una ejecución para 16-QAM. Observando la constelación de los símbolos recibidos, los obtenidos por combinación de las señales \tilde{x}_i e \tilde{y}_i a la salida de los filtros correladores (ver Fig. 8), se hace evidente la dispersión provocada por el ruido en el receptor. El panel muestra además que se ha producido un error en la demodulación de un símbolo, presumiblemente entre los símbolos correspondientes a la tercera columna de la segunda fila de la constelación de símbolos recibidos, de un total de 95 símbolos demodulados hasta ese momento. Teniendo en cuenta que se ha utilizado codificación Gray para la definición de las constelaciones (entre símbolos adyacentes únicamente hay una diferencia en un solo bit entre los símbolos binarios que representan), eso implica que sólo habrá un único bit erróneo del total de 95 símbolos \times 4 bits/símbolo = 380 bits transmitidos. Lo anterior lleva a una tasa de error de bit (BER, *Bit Error Rate*) de $1/380 = 2,632 \cdot 10^{-3}$, tal como indica el propio panel.

La Fig. 12 muestra los resultados académicos obtenidos por el alumnado en la temática de ‘Sistemas de Comunicación Digital’, antes y después de la introducción del entrenador de comunicaciones digitales como parte del programa de



(a) Panel principal durante el análisis de la modulación QPSK



(b) Panel secundario durante el análisis de la modulación 16-QAM

Fig. 11. Paneles para el análisis de las señales del entrenador

prácticas de una asignatura de máster. Durante los cursos 2013-2014 y 2014-2015 se produce un período de transición entre el máster precedente y el que lo sustituye, donde además la asignatura se traslada del primer año al segundo del máster, por lo que no se imparte, de ahí que no existan resultados para dichos cursos. Hay que indicar que el número de estudiantes de esta asignatura es reducido, con una media de cinco estudiantes anuales. Los resultados académicos mostrados se refieren al rendimiento del alumnado ante una prueba escrita donde se solicitan respuestas cortas que buscan evaluar el aprendizaje de los principales conceptos relativos a los distintos aspectos sobre sistemas de comunicación digital cubiertos por la asignatura.

En el período previo a la introducción del entrenador, se observa que durante el último curso del mismo (2012-2013) los resultados académicos experimentan una notable mejora a pesar de no contarse con esta nueva herramienta en las sesiones prácticas. La nota media sube hasta 6,2 cuando en el pasado se encontraba generalmente por debajo del 5. Sin embargo, hay que indicar que durante ese curso, con un pico de matriculados que superaba los diez estudiantes, se contó con un grupo del alumnado especialmente competente, lo que lleva a una importante alteración de la realidad general. De hecho, durante ese curso, si bien es cierto que cerca del 30% de los estudiantes superaron la calificación de 8 sobre 10 en los temas sobre 'comunicaciones digitales' de la asignatura, el 60% de

la clase obtuvo una calificación inferior a 6, donde la mitad de este segundo grupo no consiguió siquiera alcanzar el aprobado en esa temática de la asignatura. La desviación estándar de las calificaciones para este curso 2012-2013 (ver Fig. 12) delata esta situación, donde se observó una gran disparidad en el rendimiento de los distintos estudiantes.

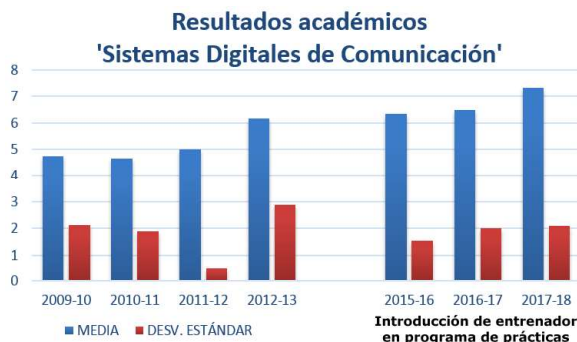


Fig. 12. Resultados académicos antes y después de incluir el entrenador como parte de las actividades prácticas

Los pobres resultados del alumnado en los temas de ‘comunicaciones digitales’ observados durante los sucesivos cursos académicos, llevan durante el año 2014 a plantearse algún tipo de solución. Es por ello que se lleva a cabo el desarrollo del entrenador de comunicaciones digitales y su implantación en el programa de prácticas de la asignatura equivalente del nuevo máster. El efecto es inmediato manifestándose en dos aspectos del rendimiento académico. Por un lado, la nota media experimenta un incremento superior al 31 %, pasando de 5,1 a 6,7. Si no tenemos en cuenta los cursos 2012-2013 y 2017-2018, donde son apreciables notas medias superiores a los promedios en ambos períodos (antes y después del uso del entrenador en las prácticas), estaríamos hablando de un incremento de un 34 % desde una nota media de 4,8 a otra de 6,4 en el nuevo período. Por tanto, es patente la mejora en los resultados promedios del alumnado. Además, el segundo aspecto a resaltar en la asignatura del nuevo máster es que todo el alumnado ha superado hasta la fecha los exámenes relativos a estos tópicos sobre ‘comunicaciones digitales’, aún cuando se enfrentaban a exámenes equivalentes a los realizados por el alumnado del anterior máster, alumnado este último donde el 40 % no fue capaz de superar ese módulo de la asignatura.

V. CONCLUSIONES

En este trabajo se ha presentado el diseño e implementación en FPGA de un entrenador para el aprendizaje de comunicaciones digitales, donde se ha hecho uso de la herramienta de programación LabVIEW para el desarrollo del *software* residente en el ordenador. Igualmente, se muestran las posibilidades ofrecidas por el prototipo en cuanto a las diferentes características de los sistemas de comunicación digital que pueden ser analizadas con el mismo. A la vista de los resultados presentados, se puede considerar que el entrenador desarrollado constituye una herramienta bastante versátil para promover una mejor comprensión de este tipo de sistemas

de comunicación, además de ser fácilmente adaptable para incorporar nuevas funcionalidades. Finalmente, se ha evaluado la potencialidad del entrenador para mejorar los resultados de aprendizaje sobre comunicaciones digitales en asignaturas de máster que abarquen esta materia, donde se aprecia una notable mejora en el rendimiento académico del alumnado, con incrementos superiores al 30 % en las calificaciones medias, tras la implantación de su uso dentro del programa de prácticas de dichas asignaturas.

Por otro lado, comprendiendo el diseño y la implementación del entrenador como un posible trabajo de fin de titulación, su consecución conlleva el estudio de los sistemas de comunicación, procesamiento digital de señales, así como el uso de lenguajes de descripción de *hardware* para diseño digital y la programación en LabVIEW. En ese sentido, constituye un reto interesante para estudiantes de titulaciones relacionadas con la ingeniería electrónica, que tendrían que poner en práctica muchos de los conocimientos y habilidades adquiridas durante su formación, lo que les capacitaría para enfrentarse a problemas similares o más complejos en su futuro trabajo como ingenieros/as.

REFERENCIAS

- [1] J.M. Santiago, Jr., A.S. Ali, “Educational communication system (EDUCOMM)-a training device,” in *Proceedings Frontiers in Education Conference*, Oct. 1988, pp. 142-148.
- [2] F. Ahamed, F.A. Scarpino, “An educational digital communications project using FPGAs to implement a BPSK detector,” *IEEE Trans. Educ.*, vol. 48, no. 1, pp. 191-197, Feb. 2005.
- [3] R. Serrano, J.M. Górriz, J. Ramírez, C.G. Puntonet, “Implementación de entrenadores de comunicaciones en placas PCB,” in *VIII Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica - TAAE 2008*, Julio 2008, p. 112.
- [4] O. González, A. Ayala, S. Rodríguez, B.R. Mendoza, “Implementación en FPGA de un bucle Costas para la demodulación de señales de AM DSB-SC,” in *XXIV Simposium Nacional de la Unión Científica Internacional de Radio - URSI 2009*, Septiembre 2009, p. 57.
- [5] T. Mizuochi, Y. Konishi, Y. Miyata, T. Inoue, K. Onohara, S. Kametani, T. Sugihara, K. Kubo, H. Yoshida, T. Kobayashi, T. Ichikawa, “Experimental demonstration of concatenated LDPC and RS codes by FPGAs emulation,” *IEEE Photonics Technol. Lett.*, vol. 21, no. 18, pp. 1302-1304, Sept. 2009.
- [6] J. Lagos-Benites, M. Grosso, M. Sonza Reorda, G. Audisio, M. Pipponzi, M. Sabatini, V.A. Avantaggiati, “An FPGA-emulation-based platform for characterization of digital baseband communication systems,” in *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*, Oct. 2011, pp. 391-398.
- [7] O. Font-Bach, N. Bartzoudis, A. Pascual-Iserte, D. López Bueno, “A real-time FPGA-based implementation of a high-performance MIMO-OFDM transceiver featuring a closed-loop communication scheme,” in *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct. 2012, pp. 100-107.
- [8] L.W. Couch II, *Digital and Analog Communication Systems*, Ed. Prentice Hall, Upper Saddle River, New Jersey, United States, 2007.
- [9] *Spartan-3A FPGA Family: Data Sheet*, Xilinx, Inc., 2010.