

Implementación en FPGA de modulador BPSK con parámetros regulables

FPGA implementation of a BPSK modulator with adjustable parameters

Mariano Morel^{†1}, Lucas Rabioglio^{†2}, Raúl Lopresti^{†3}, M. Antonelli^{†4} y Luciana De Micco^{†4}

[†]Instituto de Investigaciones Científicas y Tecnológicas en Electrónica (ICYTE)
 Facultad de Ingeniería, Universidad Nacional de Mar del Plata (FI-UNMDP)
 Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

¹marianomorel@fi.mdp.edu.ar

²lucas.rabioglio@fi.mdp.edu.ar

³lopresti@fi.mdp.edu.ar

⁴maxanto@fi.mdp.edu.ar

⁵ldemicco@fi.mdp.edu.ar

Recibido: 01/11/22; Aceptado: 01/12/22

Resumen—El trabajo presenta el diseño e implementación de un generador de señales moduladas en fase, el cual permite variar en un amplio rango sus parámetros de transmisión. Se buscó que el circuito final utilice una mínima cantidad de recursos de la FPGA. Esto fue posible ya que el diseño desarrollado almacena un cuarto de ciclo de la portadora, luego la lógica de control decide cuáles muestras tomar para obtener la frecuencia y cantidad de ciclos por dato establecidos, permitiendo una gran variación. También es configurable la potencia de salida de la señal. Además, el sistema desarrollado permite recibir los datos a través de una entrada externa o los genera en forma interna mediante un generador de números pseudo aleatorios (PRNG). El circuito se implementó mediante una FPGA (Field Programmable Gate Array) y un conversor digital analógico (DAC).

Palabras clave: Modulación de fase, FPGA, DAC, BPSK, PRNG.

Abstract—This work presents the design and implementation of a phase-modulated signal generator, which allows a wide range of variation in its transmission parameters. The implemented circuit uses a minimum amount of FPGA resources. This was possible since the developed design stores a quarter cycle of the carrier, then the control logic decides which samples to take to obtain the established frequency and number of cycles per data, allowing a great variation. The output power of the signal is also configurable. In addition, the developed system allows data to be received through an external input or generated internally through a pseudo-random number generator (PRNG). The circuit was implemented using an FPGA (Field Programmable Gate Array) and a digital-analogue converter (DAC).

Keywords: Phase modulation, FPGA, DAC, BPSK, PRNG.

I. INTRODUCCIÓN

Este trabajo es parte de un proyecto mayor, en el que se pretende desarrollar sensores espectrales para Radio Cognitiva [1]. Estos sensores deben ser capaces de detectar la presencia de señal modulada en situaciones de relación de señal a ruido muy bajas. Un punto importante es que estos sensores sean capaces de trabajar bajo distintas características de transmisión, por ello es vital el testeado de los mismos bajo múltiples escenarios. En este contexto, es

necesario contar con generadores de señales moduladas que permita realizar variaciones muy finas en cuanto a potencia, frecuencia de portadora, velocidad de datos, entre otras.

Existen trabajos en los que se proponen diseños en FPGA de moduladores BPSK (Binary Phase Shift Keying). En estos trabajos no se encontraron implementaciones con flexibilidad en los parámetros de la transmisión, tales como variación de frecuencia de portadora, cantidad de ciclos de portadora por dato, frecuencia de dato y potencia de salida.

En [2] los autores realizan la simulación de la parte digital de un transmisor BPSK. Allí se almacenan en LUTs (Look Up Tables) 13 muestras de un período de una señal senoidal, y obtienen una frecuencia de portadora de 50 MHz. Proponen enviar la señal modulada digital en forma serie mediante Serial Peripheral Interface (SPI) al DAC embebido en la placa de desarrollo Spartan 3. Esto limita la máxima frecuencia de portadora.

En [3] los autores utilizan el entorno System Generator en Matlab de la empresa Xilinx para hacer el diseño de un modulador BPSK. La señal de portadora es implementada mediante un bloque Direct Digital Synthesis Compiler, almacenan en dos memorias ROMs (Read-Only Memory) dos ciclos de señal senoidal en contra-fase. La implementación la realizan en dos placas Spartan 3E Starter Kit. Obtienen una frecuencia de portadora 31,250 kHz.

En [4] los autores almacenan en LUTs 256 muestras de un ciclo completo de una senoidal. Para obtener los defasajes según el dato a transmitir invierten el bit más significativo de la dirección. De esa forma recorren la memoria en forma directa para enviar un uno, y en forma inversa para transmitir un cero.

En [5] el diseño es también realizado mediante el entorno System Generator. Para la implementación utilizan una FPGA Virtex-4 de Xilinx. Generan la portadora almacenando 20 muestras de un ciclo de senoidal en una memoria ROM que actúa como LUT, cada dato se representa con 16 bits en aritmética de punto fijo con signo (14 bits fraccionarios). La cantidad de ciclos por dato es fija en uno, con una portadora de 5 MHz.

En este trabajo se implementa un modulador BPSK que es capaz de variar en un amplio rango la frecuencia de portadora, la cantidad de ciclos de portadora por dato y la potencia de salida. La implementación desarrollada utiliza una mínima cantidad de recursos gracias a la arquitectura mejorada que aprovecha el almacenamiento interno de un cuarto de señal senoidal para generar un ciclo completo de portadora. A su vez, se desarrolla un bloque de control de lectura de la memoria, el cual recibe los valores de los parámetros externos fijados por el usuario y en base a ellos realiza la lectura de los datos en memoria.

El resto del artículo se organiza de la siguiente forma: en la Sección II se presenta brevemente el diseño realizado, el cual permite alcanzar los requerimiento solicitados. En la Sección III se describe el circuito implementado, tanto el bloque digital como el analógico. En la Sección IV se muestran los resultados obtenidos a partir de las mediciones y pruebas realizadas. Finalmente en la Sección V se comentan las conclusiones.

II. DISEÑO DEL MODULADOR

En la Fig. 1 se muestra un esquema simplificado del diseño. Éste se divide en dos bloques principales, uno que realiza el procesamiento digital y el otro el analógico. El usuario puede elegir entre modular la portadora con datos externos o que éstos se generen internamente mediante un PRNG, esto último se implementa dentro del bloque de procesamiento digital para ser empleado en el banco de medición para pruebas. El bloque de procesamiento digital recibe los valores de los parámetros externos establecidos por el usuario, en cuanto a frecuencia requerida, cantidad de ciclos de portadora por dato, los datos modulantes externos en caso de requerirse o la activación del PRNG. Esta configuración, sumada a los datos, ingresan al bloque Control, que determina la forma en que se leerá la memoria en la que se almacena el cuarto de ciclo de portadora. Para lograr la flexibilidad requerida, se realiza un diseño que almacena en memoria N muestras correspondientes a un cuarto de ciclo de señal de portadora, esto es 90° . Por lo cual, se consigue una resolución en grados de $90^\circ/N$, por ejemplo almacenando 256 muestras la resolución en grados será de $0,3515^\circ$. Esto permite una gran variación de la frecuencia de portadora, dado que la lógica de control de barrido de la memoria puede saltarse muestras y generar una señal de salida de alta frecuencia conservando una resolución admisible (el límite de Nyquist serían 2 muestras por período completo). Esta característica también permitiría

realizar una modulación de fase M-aria. El bloque que realiza el procesamiento analógico se encarga de generar la señal modulada de salida. Para ello se requiere recibir la señal digital y la configuración realizada por el usuario de la potencia de salida deseada.

III. CIRCUITO IMPLEMENTADO

El procesamiento digital se implementa en una FPGA Cyclone III de Altera, la cual se programa mediante el lenguaje de descripción de hardware VHDL (Very High-Speed Integrated Circuit Hardware Description Language) [6], mientras que el procesamiento analógico se desarrolla en una placa que contiene el conversor DAC, la etapa para la adecuación de señal y el filtrado final.

En la Fig. 1 se puede ver un esquema del sistema implementado. Se observan los dos bloques principales mencionados y las entradas de selección que permiten variar los parámetros de la modulación.

En una comunicación BPSK los datos a transmitir tienen una duración T_b y el período de portadora es T_c . El número de períodos de portadora por dato es N_c ($T_b = N_c \cdot T_c$). La FPGA entrega al DAC las muestras de la señal modulada a una frecuencia f_s y siendo $4 \cdot N$ la cantidad de muestras por ciclo de portadora, el período de portadora resulta $T_c = 4 \cdot N \cdot T_s$. De esta forma el tiempo de dato a transmitir $T_b = 4 \cdot N_c \cdot N \cdot T_s$, siendo $T_s = 1/f_s$. A través de entradas (F en la Fig. 2), el usuario selecciona la frecuencia de portadora deseada, esto se traduce directamente en los pasos del contador que recorre la memoria. A mayor valor de F , los pasos del contador serán más grandes y en consecuencia una mayor frecuencia de portadora. También se considera la posibilidad de que el usuario varíe externamente la cantidad de ciclos de portadora por dato por medio del parámetro N_c .

III-A. Diseño en la FPGA

Un esquema del circuito implementado en la FPGA puede verse en la Fig. 2. El sistema consiste principalmente en la lectura de una memoria en la que se tiene almacenado un cuarto de ciclo de una señal senoidal. Esta memoria se lee y entrega al DAC el dato requerido según los parámetros de entrada N_c y F de Fig. 2 y el dato a transmitir. Una lógica de control se encarga de establecer si el contador que recorre la memoria realiza una cuenta ascendente o descendente, el valor inicial de esta cuenta y los pasos de incremento/decremento. También determina si el valor de la memoria debe ser negado o no. Por ejemplo, cuando se desee entregar un período senoidal a la mínima frecuencia de salida, se leerá la memoria de la siguiente forma:

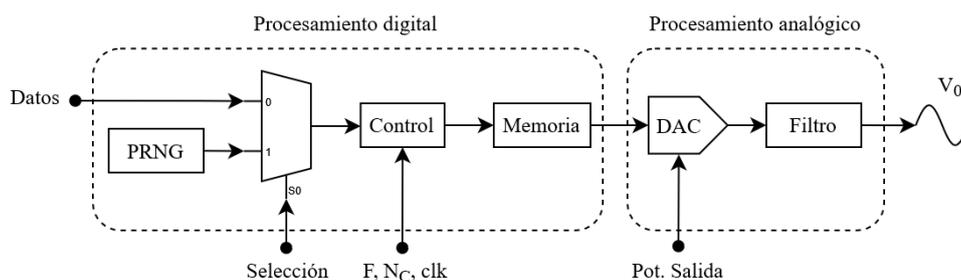


Figura 1: Esquema del diseño desarrollado.

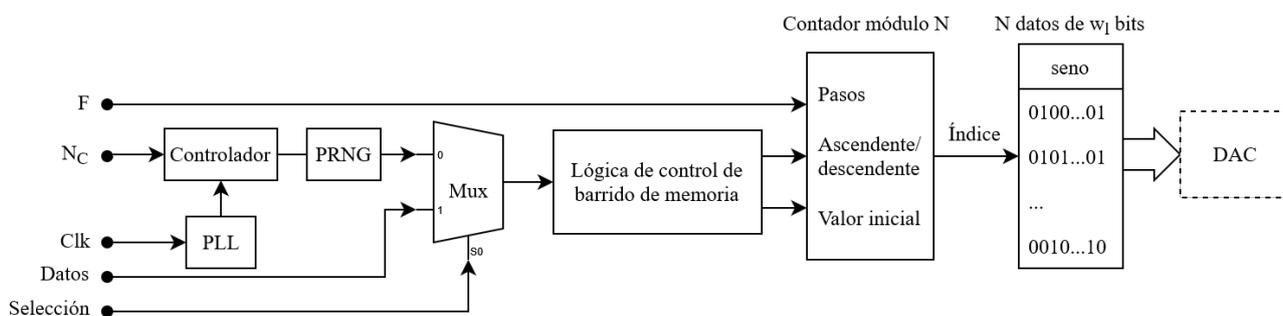


Figura 2: Esquema del circuito digital implementado en la FPGA.

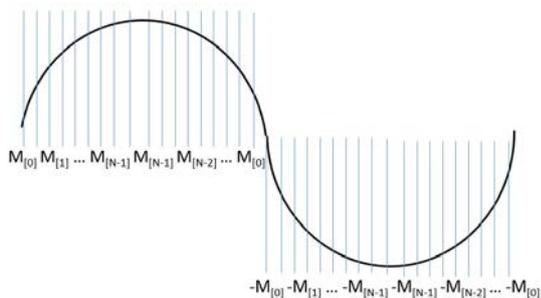


Figura 3: Generación de un ciclo de portadora.

1. Cuenta ascendente de 0 a $N - 1$, en pasos de 1.
2. Cuenta descendente de $N - 1$ a 0, en pasos de 1.
3. Cuenta ascendente de 0 a $N - 1$, en pasos de 1 y salida negada.
4. Cuenta descendente de $N - 1$ a 0, en pasos de 1 y salida negada.

Esto genera $4.N$ muestras de un ciclo de portadora, a la mínima frecuencia posible, que es $f_c = f_s/(4.N)$. A diferencia de trabajos existentes en los que usualmente se almacena un pequeño número de muestras por ciclo de portadora, como en [2] (13 muestras por ciclo) y [5] (20 muestras por ciclo), en esta implementación se almacenan $N = 256$ datos por cuarto de ciclo, o 1024 muestras por ciclo. Esta mayor cantidad de muestras permite una amplia variación de la frecuencia de portadora a la salida. La entrada de Selección determina si los datos modulantes son generados internamente mediante un PRNG, que en este caso se trata de un LFSR (Linear Feedback Shift Register) o se reciben de forma externa.

En la Fig. 3 se visualiza cómo sería la lectura de la memoria de tamaño N para la generación de un ciclo de portadora completo.

A continuación se realiza una breve descripción de cada uno de los bloques de la Fig. 2:

- Contador: Este bloque es el encargado de permitir seleccionar la cantidad de periodos de la señal portadora por cada dato de la señal modulante. Este componente posee tres entradas, una de reset, otra de reloj proveniente del bloque que implementa la Lógica de control de barrido de la memoria ROM y una tercer entrada que permite seleccionar entre 1, 2, 4 u 8 ciclos de portadora por dato mediante dos switches de la placa de desarrollo. La única salida de este bloque actúa

como entrada de reloj del bloque PRNG Datos.

- PLL: Este elemento del diseño es el encargado de generar dos señales de reloj a su salida que alimentan al resto de los componentes. Dispone de dos entradas, una de reset y otra de reloj proveniente de la placa de desarrollo, y también cuenta con tres salidas, dos salidas de reloj utilizadas por el resto del sistema y una salida adicional que permite determinar el momento en que el PLL se encuentra estabilizado. Esta señal se conectó a un diodo LED.
- PRNG: Esta parte del diseño es la encargada de generar una secuencia de datos binarios, la cuál puede ser utilizada por el sistema como señal modulante dependiendo de la selección que se realice. Además dispone de dos entradas, una de reloj y una de reset, y una salida que consiste en la secuencia generada.
- Mux 2a1: Este multiplexor permite seleccionar la procedencia de la secuencia binaria que se utiliza como señal modulante. Por un lado es posible elegir una secuencia pseudoaleatoria proveniente del bloque PRNG Datos, y por otro lado es posible seleccionar una secuencia externa.
- Lógica de control de barrido de memoria: Este componente del diseño es el encargado de realizar la lectura de la memoria ROM en función de la secuencia binaria a su entrada.
- Contador módulo N : Este elemento trabaja en conjunto con el bloque Lógica de control de barrido de memoria. Se encarga de generar la dirección de memoria necesaria en función del parámetro F de la Fig. 2 y de señales provenientes de la lógica de control.
- Memoria ROM: En la memoria ROM se almacenan 256 muestras de 11 bits que representan un cuarto de ciclo de una onda senoidal. El MSB, que representa el bit de signo, es generado por la lógica de control de forma tal que el dato enviado al DAC es el adecuado.

En la Tabla I se muestran los recursos empleados en una FPGA Cyclone III de Altera para el circuito implementado. Se puede ver que, a pesar de almacenar un gran número de muestras de portadora (lo que permite variar los parámetros mencionados), consume muy pocos recursos gracias al diseño desarrollado.

El diseño propuesto admite una frecuencia máxima de operación (f_s) de 181,62 MHz lo que permitiría una máxima frecuencia de portadora de 90,81 MHz, y una mínima de 177 kHz. En la Fig. 4 se puede ver cómo varía la frecuencia

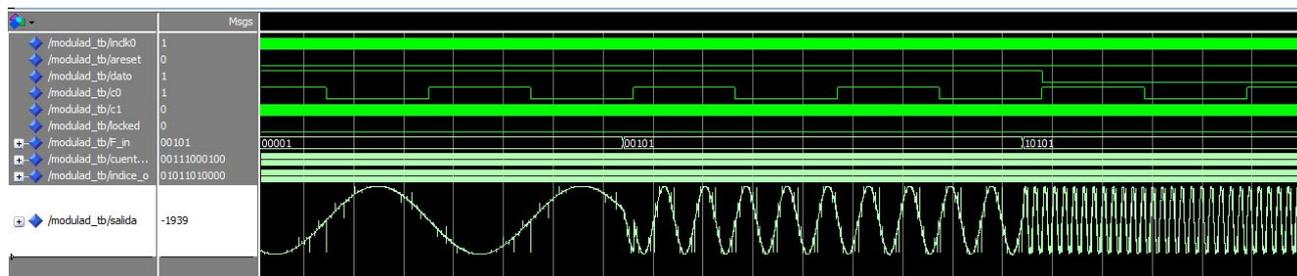


Figura 4: Simulación mediante ModelSim de la salida ante la variación de la frecuencia de portadora.

Cuadro I: Recursos empleados en la implementación utilizando una FPGA Cyclone III de Altera.

Elementos lógicos (EL) totales (119.088)	655
Combinacional de EL	518
Registros de EL	44
Registros dedicados(121.673)	137
PLL	1
f_{Max} [MHz]	181,62

de portadora ante el incremento de la entrada F (en la figura llamada F_{in}).

En la Fig. 5 se puede ver la señal generada por el circuito, la señal de portadora (en amarillo) varía su fase según los datos de entrada (en celeste).

III-B. Procesamiento analógico

Esta parte del sistema se encarga de generar una señal analógica a partir de la señal digital recibida desde la FPGA. El diseño de este bloque se basa en el CI BA9221 [7], el cuál es un convertor Digital-Analógico de 12 bits que permite trabajar en complemento a dos. El CI BA9221 es un convertor con salida de corriente, la cuál es proporcional a una corriente de referencia. A la salida del CI BA9221 se utiliza una etapa de adecuación de señal para convertir la corriente de salida del DAC en tensión. Para lograr esto, se usa el CI NE5532 que consiste en un operacional doble. Así mismo, mediante un potenciómetro, es posible variar la corriente de referencia del DAC y de esta forma lograr variar la tensión de salida. La máxima tensión de salida es de $10V_{pp}$. En la Fig. 6 se aprecia un diagrama en bloques de la parte del convertor Digital-Analógico. Además, desde la FPGA se debe efectuar la negación del bit mas significativo (MSB) del bus de datos que ingresa al CI BA9221, de forma

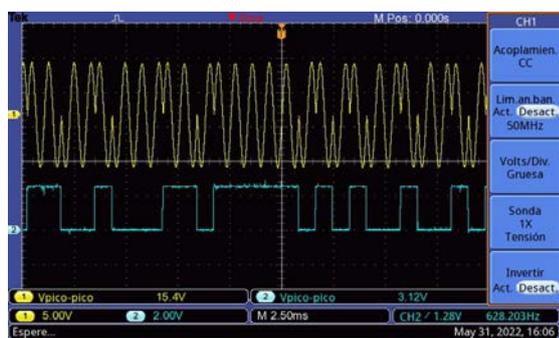


Figura 5: Captura de osciloscopio. Datos modulantes y señal modulada.

tal de poder trabajar de forma adecuada en complemento a dos.

Por otro lado, a continuación de la etapa de conversión, se diseña un filtro pasabajos RC de primer orden con la finalidad de reducir el ruido de cuantificación proveniente de la etapa anterior. El diseño de la frecuencia de corte de este filtro se contempla para que se encuentre aproximadamente una década por encima de la máxima frecuencia de portadora, resultando de esta manera una $f_{corte} \approx 284 kHz$.

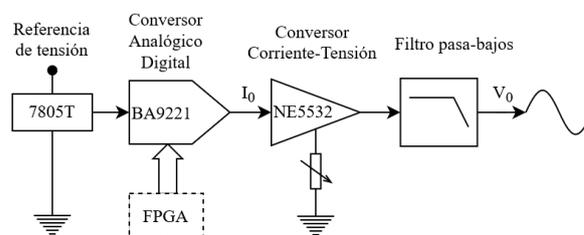


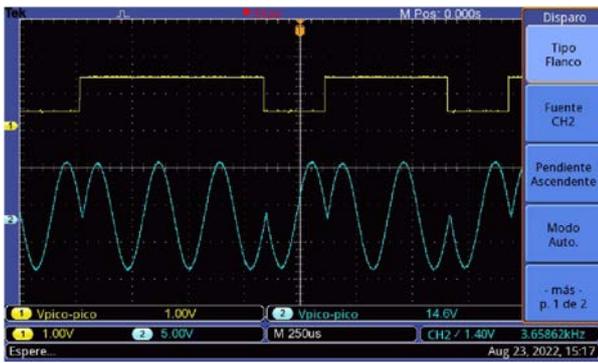
Figura 6: Circuito esquemático del convertor DAC.

IV. RESULTADOS

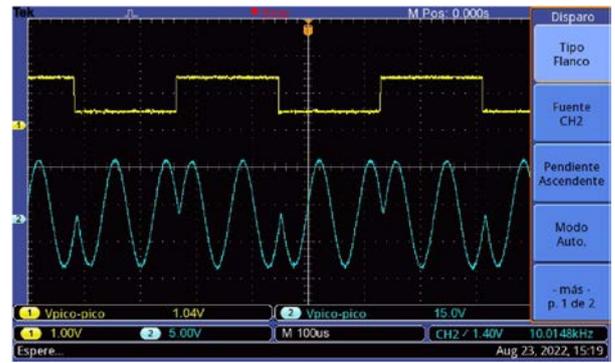
En esta sección se detallan los resultados obtenidos de la implementación del diseño propuesto. Se realizan varias capturas de osciloscopio para diferentes parámetros de entrada que permitan verificar el funcionamiento del sistema en su conjunto. En las Figuras 7a, 7b, 7c y 7d se observan distintas capturas de la señal de salida modulada y de la señal de entrada para diferentes valores de los parámetros de entrada F y N_c de la Fig. 1. Por medio del parámetro F es posible cambiar la frecuencia de la señal portadora. Y el parámetro N_c permite determinar la cantidad de ciclos de portadora por dato binario de entrada. En las Figuras 7a, 7b, 7c y 7d se visualiza que la frecuencia de la señal modulada difiere mínimamente de lo planteado en el diseño. Además es posible advertir un reducido desfase entre la señal modulante y la señal modulada, lo cuál es producido por el filtro pasabajos a la salida del bloque DAC de la Fig. 1.

V. CONCLUSIONES

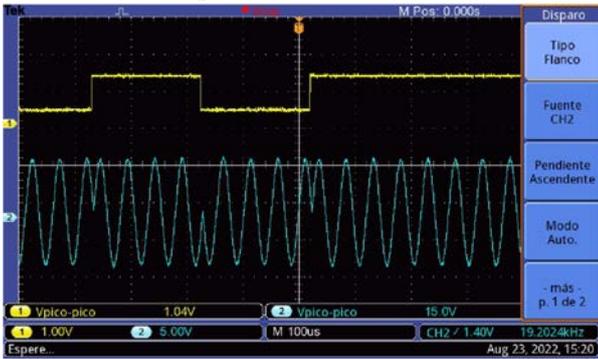
En este trabajo se diseñó e implementó un generador de señal modulada en fase. El diseño implementado permite variar los parámetros de la señal modulada de salida como la frecuencia de portadora, potencia de salida, etc. La cantidad de recursos empleados en la FPGA es mínimo gracias a la utilización de una memoria que almacena un cuarto de ciclo de una forma de onda senoidal, la cual es leída mediante una lógica de control para generar la señal de portadora. También



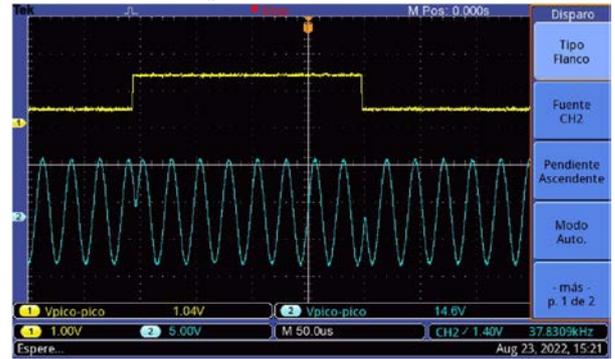
(a) Captura con $F = 3$ y $N_c = 1$



(b) Captura con $F = 9$ y $N_c = 2$



(c) Captura con $F = 17$ y $N_c = 4$



(d) Captura con $F = 32$ y $N_c = 8$

Figura 7: Capturas de osciloscopio en donde se muestra la variación de f_s y cantidad de ciclos de portadora por dato.

se diseñó e implementó la placa que recibe la señal digital y entrega la señal analógica, la cual permite variar la potencia de salida de la misma.

AGRADECIMIENTOS

Este trabajo fue parcialmente financiado por CONICET, Agencia I+D+i (PICT19 3024), la FI-UNMdP, Argentina y los programas ICTP associates e IAEA/ICTP STEP.

REFERENCIAS

[1] Y. Arjoune and N. Kaabouch, "A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions," *Sensors (Switzerland)*, vol. 19, no. 1, 2019.

[2] A. K. Chaudhary, P. Pratik, S. Gupta, and V. Kakkar, "Improved digital design of bpsk modulator using look-up table technique," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2013, pp. 47–52.

[3] S. Popescu, A. Gontean, and G. Budura, "Bpsk system on spartan 3e fpga," in *2012 IEEE 10th International Symposium on Applied Machine Intelligence and Informatics (SAMII)*. IEEE, 2012, pp. 301–306.

[4] A. Al Safi and B. Bazuin, "Fpga based implementation of bpsk and qpsk modulators using address reverse accumulators," in *2016 IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 2016, pp. 1–6.

[5] Y. Chye, M. Ain, and N. M. Zawawi, "Design of bpsk transmitter using fpga with dac," in *2009 IEEE 9th Malaysia International Conference on Communications (MICC)*. IEEE, 2009, pp. 451–456.

[6] M. Morel, "GitHub." GitHub repository, 2022. [Online]. Available: <https://acortar.link/2RpRLF> (accedido el 1 de julio de 2022).

[7] Ba9221. [Online]. Available: <http://www.synfo.nl/datasheets/BA9221.pdf>