

# DESARROLLO DE UN SISTEMA PARA EL CONTEO DE MÚLTIPLES SEÑALES EN COINCIDENCIA BASADO EN TECNOLOGÍA FPGA

## DEVELOPMENT OF AN FPGA-BASED COINCIDENCE COUNTING SYSTEM FOR MULTIPLE SIGNALS

M. Bolaños<sup>\*1</sup>, A. Boette<sup>2</sup>, F. Videla<sup>1,3</sup> y L. Rebón<sup>2</sup>

<sup>1</sup>Centro de Investigaciones Ópticas (CIOp) – Universidad Nacional de La Plata – CIC – CONICET – Argentina

<sup>2</sup>Instituto de Física de La Plata (IFLP), Universidad Nacional de La Plata – CONICET – Argentina

<sup>3</sup>Departamento Ciencias Básicas, Facultad de Ingeniería UNLP Avenida 1 y 47, La Plata, Argentina

Recibido: 07/07/2022; Aceptado: 12/01/2023

La detección de señales en coincidencia desempeña un rol importante en distintas aplicaciones tecnológicas de metrología o comunicación, y en particular en el campo de la óptica cuántica. En este sentido, la detección simultánea de dos o más fotones permite explotar las características no clásicas de la luz, las cuales se manifiestan mediante correlaciones que no son posibles en una descripción clásica. Presentamos aquí el desarrollo de módulos programables, basados en tecnología FPGA, que componen el sistema de detección y conteo de pulsos en coincidencias. La arquitectura implementada en una placa de desarrollo de la firma Xilinx, permite determinar cuándo dos o más pulsos coinciden dentro de una ventana temporal, cuyo ancho puede configurarse entre 1 y 255 ns para compensar posibles diferencias de camino. Como producto de las medidas de pulsos individuales y en coincidencias, se obtuvieron y compararon las estadísticas de detección con las predichas teóricamente. Adicionalmente, se emularon las señales correspondientes a diferentes fuentes de fotones a través de un divisor de haz, y se obtuvieron las estadísticas de fotodetección a partir de las cuales se determinó, para cada caso, la función de correlación de segundo orden  $g^{(2)}(\tau)$ .

*Palabras clave:* FPGA, Estadística de Fotodetección, Medidas de Correlación.

The detection of temporally coincident signals plays an important role in different technological applications for metrology and communication, and particularly, in the area of quantum optics. In this sense, the simultaneous detection of two or more photons allows one to take advantage of the non-classical features of light that manifest themselves in certain types of correlations that cannot be described classically. We present here the development of programmable modules, by using FPGA technology, which are part of the coincident pulse detection and counting system. Our implementation, in a development board from Xilinx, allows to determine when two or more pulses overlap within a temporal window, with a configurable width in the range of 1 and 255 ns to compensate for possible path differences. We obtained the photodetection statistics result of the single and coincident pulse measurements, and compared them with that predicted by a theoretical analysis. In addition, we have emulated the signals corresponding to different photon sources through a beam splitter, and obtained the photodetection statistics from which we determine, for each case, the second order correlation function  $g^{(2)}(\tau)$ .

*Keywords:* Field Programmable Gate Array (FPGA), Photodetection Statistics, Correlation Measurements.

<https://doi.org/10.31527/analesafa.2023.34.1.27>



ISSN 1850-1168 (online)

### I. INTRODUCCIÓN

El desarrollo de las tecnologías cuánticas, y en particular los avances en el campo de la óptica cuántica [1], nos permiten en la actualidad generar, manipular y detectar fotones de a uno. En este contexto, el desarrollo de módulos contadores de coincidencias resulta fundamental para explotar las características no clásicas de ciertos estados de la luz, entre las que se destaca el fenómeno de entrelazamiento [2]. Este tipo de dispositivos permite realizar medidas de detección simultánea de dos o más fotones, requerimiento para muchas aplicaciones de la óptica cuántica, como por ejemplo en criptografía cuántica [3] o metrología cuántica [4].

En este trabajo, presentamos el diseño y desarrollo de un módulo para el conteo de señales en coincidencia en un circuito integrado FPGA (Field Programmable Gate Arrays),

que consiste de un arreglo matricial de bloques lógicos programables e interconectados [5]. El funcionamiento de nuestra implementación está asociado al de fotodetectores basados en fotodiodos de avalancha (APDs) o fotodiodos de avalancha para fotones individuales (SPADs), los que permiten detectar señales de muy baja intensidad, del orden del  $\text{pW}/\text{cm}^2$  [6, 7], y en particular pueden emplearse para detectar fotones de a uno al ser operados en modo Geiger.

Dado un conjunto de señales de entrada (por defecto cuatro), nuestro módulo es capaz de contar el número de arribos de dos o más de estas señales en coincidencia (y al mismo tiempo los arribos individuales) dentro de ventanas temporales cuyos anchos pueden ser ajustados entre 1 y 255 ns. Con el objetivo de caracterizar el sistema se implementó, en la misma placa de desarrollo, un generador de pulsos de período configurable. Se generaron trenes de pulsos con distintos períodos y se midieron cuentas en coincidencia utili-

\* matiasruben.bolanoswagner@studenti.unipd.it

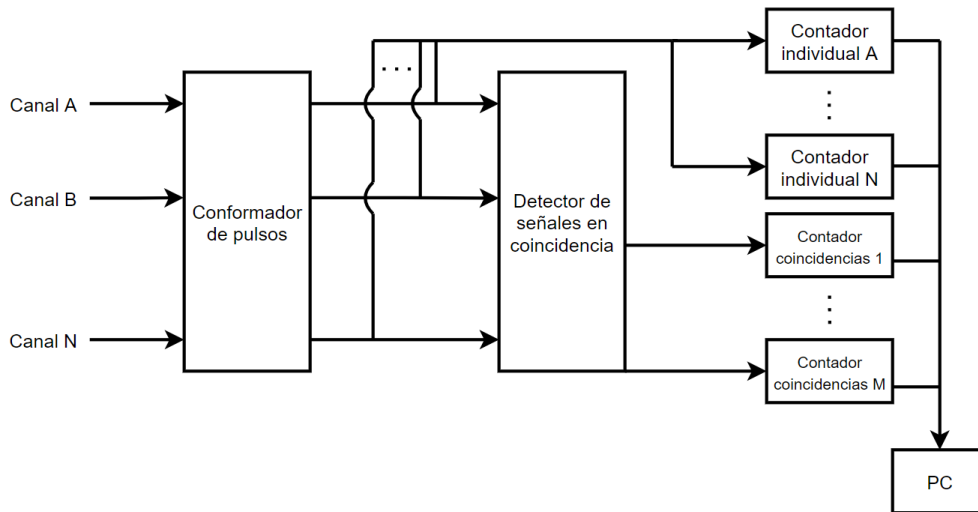


FIG. 1: Diagrama en bloques del módulo contador de coincidencias. Las  $N$  señales ingresan por los canales de entrada, son retardadas individualmente, y los pulsos son conformados a un mismo ancho configurable.

zando el módulo desarrollado, observándose que los resultados están en acuerdo con lo esperado. En segunda instancia, se implementaron trenes de pulsos aleatorios que simulan el decaimiento de una muestra radiactiva y se analizó el registro de las cuentas individuales y en coincidencia, donde se encontró también que los resultados coinciden con lo esperado para este tipo de fuentes. Finalmente, se implementó un sistema capaz de, dado un estado inicial configurable por computadora e ingresado como parámetro, realizar una simulación del experimento de Hanbury-Brown-Twiss (o HBT) para calcular la función de correlación de segundo orden  $g^{(2)}(\tau)$  [8]. Este último módulo se utilizó para simular el comportamiento de una fuente láser y de una fuente de estados de fotones individuales.

## II. DISEÑO DEL CIRCUITO

El objetivo de este trabajo fue el diseño de un sistema capaz de contar la cantidad de veces que un conjunto de señales de entrada coinciden en un intervalo de tiempo determinado (ventana de coincidencias). La detección de coincidencias implica la introducción de retardos configurables en el orden de los nanosegundos o menor, por lo que empleamos el retardo propio de los elementos básicos de la tecnología Xilinx. En general, en los circuitos digitales se implementan retardos basados en cadenas de *flip-flops* y su fiabilidad reposa en el empleo de relojes generados por cristales considerados estables. Sin embargo los retardos de estas compuertas, a pesar de su naturaleza analógica, puede verificarse que son controlables. Así, partiendo de estas consideraciones generales, se deducen los criterios de diseño y las posibilidades de VHDL como lenguaje descriptor de esta implementación, un poco apartada de los diseños estándar. Otros módulos que componen este desarrollo siguen los criterios usuales (contadores, memorias, UART's) aunque el desarrollo de los protocolos de comunicación con la interfase de usuario está diseñado con un formato *ad-hoc*.

El sistema considerado tiene antecedentes en los trabajos [9-11], y su diagrama de bloques se muestra en la Fig. 1. Su funcionamiento puede describirse según las siguientes

etapas:

- Los pulsos de las  $N$  entradas, que pueden provenir de un fotodetector o un generador de señales, ingresan a un conformador de pulsos, que transforma los pulsos entrantes en unos rectangulares con ancho temporal configurable.
- Un módulo detector a dónde ingresan los pulsos ya conformados. El mismo está basado en un conjunto de compuertas AND que permiten registrar la superposición temporal de las señales. La ocurrencia de solapamiento implica la detección de una coincidencia entre el subconjunto de señales seleccionado. Como resultado de la coincidencia el circuito emite un pulso. Agrupando las entradas se dispone de  $M$  configuraciones de coincidencias dando lugar a  $M$  salidas, asociadas a las diferentes configuraciones.
- El bloque de contadores independientes cuenta los pulsos registrados en una ventana temporal de ancho programable. Los pulsos a contar provienen de los  $N$  canales de entrada, y de las  $M$  salidas en coincidencia. Esta información es luego enviada a una PC por medio de una interfaz implementada en un módulo UART que se incorporó en la propia FPGA.

### Implementación en FPGA

Estos sistemas se implementaron sobre un chip FPGA ubicado en una placa de desarrollo de prototipos Xilinx Spartan-3A. Un chip FPGA puede ser programado utilizando lenguajes de descripción de hardware como VHDL (utilizado en este trabajo) o Verilog. Para facilitar el desarrollo, el sistema completo se dividió en un conjunto de submódulos individuales o componentes, que luego se conectaron a través de un código integrador. Todos los componentes fueron desarrollados en función de parámetros globales como la cantidad de canales de entrada ( $N$ ) y la cantidad de canales de salida ( $M$ ) para facilitar la escalabilidad y adaptabilidad del diseño. Todos los archivos de código se encuentran en [12], requiriendo petición de acceso.

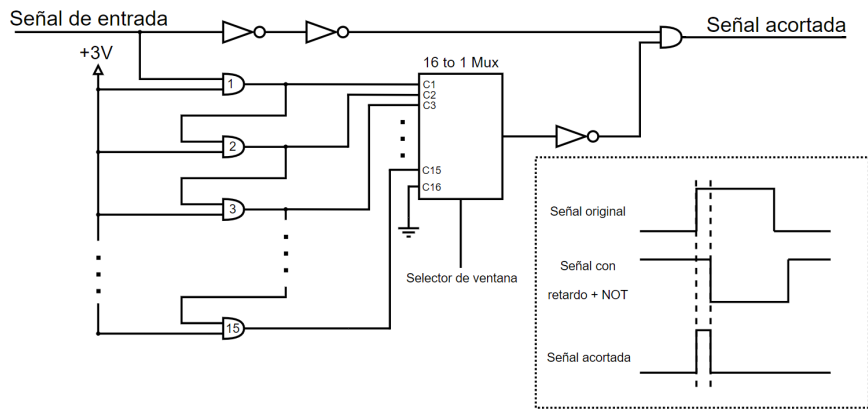


FIG. 2: Circuito reducido (solo 16 compuertas) implementado para conformar un pulso de entrada a un ancho configurable.

### Retardador

El módulo retardador es el encargado de adicionar un desfase temporal a una o más señales de entrada. Esto se vuelve necesario cuando las señales que arriban al circuito han experimentado una diferencia de caminos y se necesita compensarla, o cuando se requiere un desfase temporal particular entre ellas, por ejemplo para obtener medidas de correlación temporal. La implementación de los retardos en FPGA puede realizarse mediante recursos analógicos, con una cadena de compuertas AND, o digitales, utilizando cadenas de *flip-flops* disparados secuencialmente por flancos de la señal de reloj. La cadena de compuertas AND provoca un retardo efectivo, igual a la suma de los retardos individuales inherentes a toda implementación circuital, el cual depende de parámetros de diseño geométricos y tecnológicos. Estos retardos individuales producidos por cada compuerta, se pueden considerar despreciables para aplicaciones generales, pero son apreciables en el caso de conteo de fotones.

Para implementar el módulo retardador, se implementó una combinación de ajustes gruesos, utilizando cadenas de flip-flops controladas por la señal de reloj, y finos, utilizando una cadena de compuertas AND.

### Conformador de pulsos

El módulo conformador de pulsos tiene la finalidad de reducir el ancho de un pulso entrante. Ese ancho se puede reducir en forma gradual por medio de un parámetro de configuración cuyo valor representa el tiempo en que la señal será retardada. Esta señal es luego invertida, y enviada junto a la señal original a una compuerta AND (Fig. 2). Para implementar el retardo de la señal, con resoluciones de hasta algunos nanosegundos, se utilizó una cadena de compuertas AND como la propuesta en la sección anterior. La máxima longitud de esta cadena se fijó en 255 compuertas.

La señal ingresa a una cadena de compuertas AND en cuyas entradas, en el nivel  $i$ , se tiene una señal de 3 V y la señal saliente de la compuerta  $i - 1$ . En la compuerta 1 ingresará la señal original. Las salidas de las 255 compuertas, junto con una señal conectada a tierra, son enviadas a un multiplexor 256 a 1. La salida se selecciona con una palabra de 8 bits. A la entrada seleccionada en el multiplexor se le

**Algoritmo 1:** Pseudocódigo para acortar un pulso utilizando una cadena de compuertas AND y un selector de 8 bits. La doble operación NOT sobre la señal `salida_auxiliar(0)` está presente para compensar los tiempos de actuación del resto de los componentes, de forma que el retardo sea solo debido a las compuertas AND.

```

salida_auxiliar(0) <= pulso de entrada;
pulso acortado <= NOT NOT salida_auxiliar(0)
AND NOT salida_auxiliar(selector);
for ( $i = 0$ ;  $i < 256$ ;  $i++$ ) do
| salida_auxiliar( $i + 1$ ) <= salida_auxiliar( $i$ ) AND
| '1';
end

```

aplica una compuerta NOT, y se la ingresa a una compuerta AND junto con la señal original y luego de haberle aplicado dos compuertas NOT (Algoritmo 1). En el Inserto de la Fig. 2 se detalla el diagrama de tiempos y se esquematiza cómo actúa el circuito sobre un dado pulso. La señal original es retardada e invertida. Al ingresar junto con la señal original a una compuerta AND, se obtiene una versión del mismo pulso con igual flanco de subida que la señal original, pero acortado.

### III. CARACTERIZACIÓN Y RESULTADOS

Para ensayar y analizar el correcto funcionamiento del sistema, en primera instancia se implementó un módulo generador de pulsos periódicos de período configurable y 20 ns de ancho. De esta forma, se utilizaron pulsos de 40, 60, 100 y 140 ns para los canales de entrada A, B, C y D respectivamente, y se registraron las cuentas en coincidencia para todas las combinaciones de dos canales, en un tiempo de integración igual a 5 ms. En la Tabla 1 se reportan el número de cuentas individuales registradas en cada canal y el número de cuentas en coincidencia, y se comparan con las esperadas idealmente.

En segunda instancia, se implementó un generador de pulsos tal que los intervalos de tiempo entre pulsos sean aleatorios, lo que simula, por ejemplo, el decaimiento de una muestra radiactiva (Algoritmo 2). La inspección de un histograma de cuentas registradas para un canal revela que

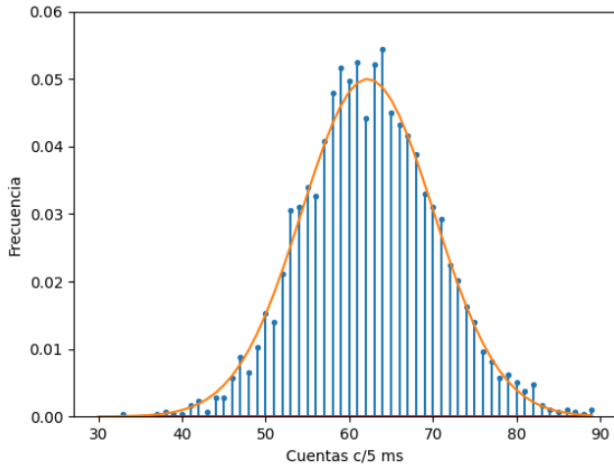


FIG. 3: Histograma de las cuentas registradas en 5 ms para el generador de pulsos aleatorios. En naranja, el ajuste correspondiente es consistente con la distribución Poissoniana esperada.

**Algoritmo 2:** Pseudocódigo para simular el proceso de decaimiento radiactivo, con  $\bar{N}$  el número medio de cuentas por segundo, y cada 60 ns decide si emitir o no un pulso.

```

if detecto pulso de entrada then
   $P_{emision} = \bar{N} \cdot 60 \text{ ns}$ ;
  rng  $\leftarrow$  número aleatorio entre 0 y 1 ;
  if rng <  $P_{emision}$  then
    | emito un pulso de 20 ns;
  end
end

```

el mismo sigue una distribución de Poisson (Fig. 3), como se espera para este tipo de fenómenos [1]. Luego, se utilizaron dos de estos trenes de pulsos aleatorios independientes como entradas para los canales A y B, y se midieron las cuentas en coincidencia entre ambos. Siendo que ambos eventos son, en teoría<sup>1</sup>, independientes, si ocurre una detección en coincidencias será de origen casual. La tasa de detección en coincidencia entre los canales independientes A y B viene dada por

$$P_{AB} = P_A P_B \Delta t,$$

donde  $\Delta t$  es la ventana de coincidencias, y  $P_A$  y  $P_B$  son las tasas de generación de pulsos. Siendo que los pulsos generados por el algoritmo 2 lo hacen cada 60 ns, para estas entradas la ventana de coincidencias efectiva es de  $\Delta t_{eff} = 60$  ns, con un número medio de cuentas por segundo  $\bar{N}$ . Con los parámetros utilizados, se esperan obtener  $\bar{N} = 12500$  cuentas por segundo en cada canal individual, por lo que se espera obtener  $P_{AB} = 9.38$  cuentas/s. Para una medida de 1 minuto de duración, se obtuvo una tasa de detección en coincidencias de  $P_{AB} = 9.87(82)$  cuentas/s, que se encuentra dentro de lo esperado.

<sup>1</sup>Se asume que los números generados por el RNG implementado en la FPGA son independientes al modificar la semilla inicial. Para garantizar la falta de sesgos, se utilizó un generador cuántico de números aleatorios para generar las semillas [13].

TABLA 1: Comparación de las cuentas por segundo esperadas y las registradas para los cuatro canales de entrada y para todas las coincidencias dobles entre estos canales.

Canal	Cuentas esperadas	Cuentas registradas
A	125000	125000
B	83333.3	83333.3(2)
C	50000	50000
D	35714.28	35714.28(20)
AB	41666.6	41666.6(2)
AC	25000	25000
AD	17857.14	17857.14(20)
BC	16666.6	16666.6(2)
BD	11904.76	11904.76(18)
CD	7142.85	7142.85(18)

### Simulación del experimento de Hanbury-Brown-Twiss

El experimento de HBT (esquemático en la Fig. 4) permite obtener información sobre la naturaleza de la luz, a partir de la función de correlación de segundo orden  $g^{(2)}(\tau)$  definida como

$$g_C^{(2)}(\tau) = \frac{\langle E^*(t)E^*(t+\tau)E(t+\tau)E(t) \rangle}{\langle E^*(t)E(t) \rangle^2} = \frac{\langle I(t)I(t+\tau) \rangle}{\langle I(t) \rangle^2}, \quad (1)$$

donde los promedios se realizan sobre un modo en el tiempo. Para disponer de un sistema controlable y fácilmente configurable que permita, por un lado, probar el funcionamiento del módulo contador de coincidencias con señales realistas, y por otro lado, obtener datos compatibles con un experimento óptico, se optó por implementar una simulación del proceso de detección de una fuente de luz arbitraria en la propia FPGA, de este modo se pueden generar señales para enviar a los distintos puertos de entrada que son compatibles con una medición realista. La simulación para este experimento particular recibe una serie de parámetros calculados mediante software desde una computadora que describen las probabilidades de detectar fotones en alguno o ambos de los detectores.

Con la intención de mantener esta simulación en una escala realizable con el espacio disponible dentro del chip, se limitó el espacio de la base utilizada a aquel en el que como máximo solo puedo tener 10 fotones entre ambas entradas/salidas del divisor de haz. De esta forma, un estado puro arbitrario se representa de la forma

$$|\psi\rangle_{1,2} = \sum_{\substack{i,j \leq 10 \\ i+j \leq 10}} \alpha_{i,j} |i\rangle_1 \otimes |j\rangle_2 = \sum_{\substack{i,j \leq 10 \\ i+j \leq 10}} \alpha_{i,j} |i,j\rangle, \quad (2)$$

mientras que para un estado mixto arbitrario

$$\rho_{1,2} = \sum_{\substack{i,j,k,l \leq 10 \\ i+j \leq 10 \\ k+l \leq 10}} \alpha_{i,j,k,l} |i,j\rangle \langle k,l|, \quad (3)$$

donde  $|i\rangle_n$  representa un estado de  $i$  fotones en la entrada/salida  $n$ . Utilizando esta representación para los estados de entrada y salida del divisor de haz, se programó dentro

**Algoritmo 3:** Pseudocódigo para simular el proceso de detección en un escenario similar al experimento de HBT

```

if detecto pulso de entrada then
  rng ← número aleatorio entre 0 y 1 ;
  if rng <  $\sum_{i,j}^{n,m} p_{i,j}$  ( $n$  fotones en el detector  $T$  y  $m$ 
  en el detector  $R$ ) then
    if  $n > 0$  and  $m > 0$  then
      salida1 ← 1;
      salida2 ← 1;
    else if  $n > 0$  then
      salida1 ← 1;
      salida2 ← 0;
    else if  $m > 0$  then
      salida1 ← 0;
      salida2 ← 1;
    else
      salida1 ← 0;
      salida2 ← 0;
    end
  end
end

```

del software de la interfaz gráfica el comportamiento del estado de salida a partir de la matriz del divisor de haz. Dado un estado de entrada  $|i, j\rangle$ , la acción del operador divisor de haz  $\hat{U}_{BS}$  es tal que

$$\hat{U}_{BS}|i, j\rangle = \sum_{k=0}^j \sum_{l=0}^i \beta_{i,j,k,l} |l+k\rangle_1 |i-l+j-k\rangle_2, \quad (4)$$

donde

$$\beta_{i,j,k,l} = \frac{\binom{i}{l} \binom{j}{k} (-1)^{j-k} \left(\frac{1}{\sqrt{2}}\right)^{i+j} \sqrt{(k+l)! (i-l+j-k)!}}{\sqrt{i! j!}}, \quad (5)$$

y el estado entrante  $|i, j\rangle$  representa el estado con  $i$  fotones en la entrada 1 y  $j$  fotones en la entrada 2, y el estado resultante  $|l+k\rangle_1 |i-l+j-k\rangle_2$  representa el estado con  $l+k$  fotones en la salida 1 e  $i-l+j-k$  fotones en la salida 2.

Para definir los parámetros que se ingresan al sistema desde la computadora, dado un estado inicial (puro o mixto) definido por el usuario, se le aplica el operador  $\hat{U}_{BS}$ , y se obtiene un vector de elementos  $p_{i,j}$ , que representan la probabilidad de detectar  $i$  fotones en el detector  $T$  y  $j$  fotones en el detector  $R$  en un dado instante de tiempo. De esta forma, para simular el proceso de detección, una vez teniendo los parámetros, se utiliza un generador de números aleatorios integrado en el chip para decidir aleatoriamente en que detectores se detecta un pulso.

Utilizando el algoritmo 3, se realizó una simulación del experimento de HBT dentro del chip FPGA. La simulación se realizó utilizando una estadística de fuente láser (descrita por un estado puro coherente), y de fotón individual (descrita por el estado puro  $|1\rangle$ ). Para ello, se ingresaron al chip los parámetros que contienen la estadística de fotodetección post-divisor de haz de ambas fuentes, y se utilizó el módulo retardador de pulsos para retardar uno de ellos respecto

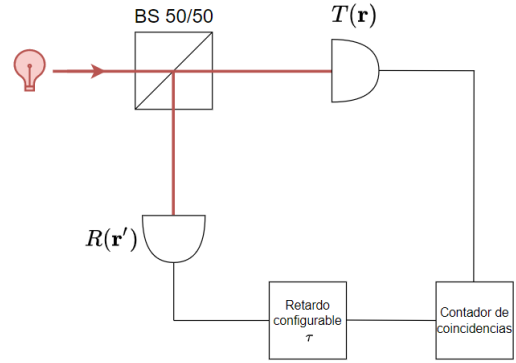


FIG. 4: Esquema del experimento de Handbury-Brown y Twiss, utilizado para determinar  $g^{(2)}(\tau)$  mediante las cuentas en coincidencias en los detectores  $T$  y  $R$ , donde el pulso medido en  $R$  es retardado un tiempo  $\tau$  configurable.

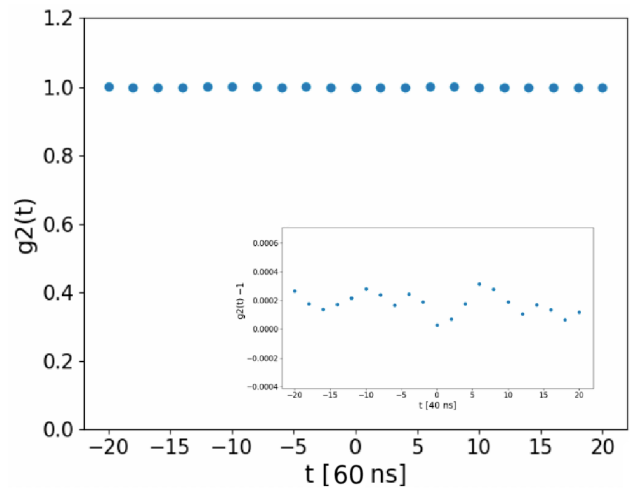


FIG. 5: Valores de  $g^{(2)}(\tau)$  obtenidos simulando una fuente tipo láser. Inserto: Ampliación en el eje de  $g^{(2)}(\tau) - 1$ , mostrando que se obtuvo  $g^{(2)}(\tau) = 1.0002(2)$ .

del otro una cantidad  $\tau$  múltiplo de 60 ns. De esta forma, se obtuvo la función de correlación de segundo orden  $g^{(2)}(\tau)$  para ambas fuentes (Fig. 5 y 6).

Para ambas fuentes simuladas, los resultados son los esperados. Para el caso de la fuente láser,  $g^{(2)}(\tau) = 1.0002(2) \forall \tau$ , que contiene el valor esperado de  $g^{(2)}(\tau) = 1 \forall \tau$ . Por otro lado, para la fuente de estados de un solo fotón se observa que para  $\tau = 0$ ,  $g^{(2)}(0) = 0$ , pero crece rápidamente a 1 para cualquier otro valor, que coincide con los resultados esperados. Usualmente, se esperarí obtener una caída continua entre  $g^{(2)} = 1$  y  $g^{(2)} = 0$ , pero al ser la fuente simulada en múltiplos enteros de 60 ns, esta continuidad no se observa, y la caída es abrupta.

#### IV. CONCLUSIONES

Se ha diseñado e implementado un sistema basado en tecnología FPGA y programado en lenguaje VHDL para registrar cuentas simples y/o en coincidencias múltiples de pulsos que ingresan al módulo de conteo por un conjunto de 4 entradas independientes. Además se ha desarrollado una interfaz gráfica que habilita el monitoreo del conteo de pulsos en tiempo real, y la reconfiguración de los parámetros de medida.

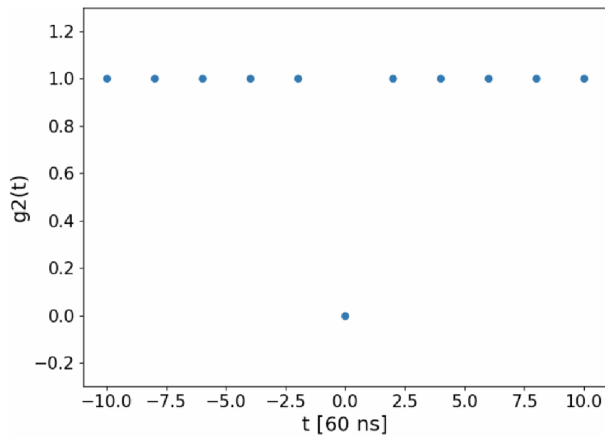


FIG. 6: Valores de  $g^{(2)}(\tau)$  obtenidos simulando una fuente de estados de Fock de un solo fotón.

Una caracterización preliminar, usando los propios recursos de diseño de los que hemos provisto al sistema, proporciona resultados aceptables en cuanto a detección de cuentas simples y en coincidencias. Se implementaron generadores de pulsos periódicos y con secuencias aleatorias cuya estadística de fotodetección verifica la esperada teóricamente. Finalmente, se implementó un módulo capaz de simular el proceso de fotodetección para fuentes con estadísticas arbitrarias configurables al interactuar con un divisor de haz 50/50. Utilizando este sistema, se simuló el experimento de Hanbury-Brown-Twiss para fuentes tipo láser y de estados de un solo fotón. Los resultados obtenidos con esta simulación son coherentes con lo esperado, constituyendo un indicador de su buen rendimiento para el análisis de las señales típicas en un experimento de óptica cuántica.

El desarrollo de este módulo forma parte de un primer prototipo para un sistema autónomo de comunicaciones cuánticas a grandes distancias. En particular, se prevé implementarlo en forma integrada a un nanosatélite, luego de modificaciones de diseño y construcción para cumplir con los requerimientos propios a la misión. Como actividad futura se continuará con la adaptación de este desarrollo a geometrías y materiales aptos para las exigencias mecánicas, térmicas y electromagnéticas inherentes a su funcionamiento en una órbita baja (LEO).

## REFERENCIAS

- [1] A. M. Fox, M. Fox *et al.* *Quantum optics: an introduction* (Oxford university press, 2006).
- [2] A. K. Ekert. en *Quantum Measurements in Optics* 413-418 (Springer, 1992).
- [3] C. H. Bennett y G. Brassard. Quantum cryptography: Public key distribution and coin tossing. [arXiv preprint arXiv:2003.06557](https://arxiv.org/abs/2003.06557) (2020).
- [4] E. Polino, M. Valeri, N. Spagnolo y F. Sciarrino. Photonic quantum metrology. *AVS Quantum Science* **2**, 024703 (2020).
- [5] P. Dillien. And the winner of best FPGA of 2016 is. *EE Times* **33** (2017).
- [6] S. A. Bordakevich, A. Kielbowicz, I. H. López Grande y M. A. Larotonda. Módulo de conteo de fotones basado en fotodiodos de avalancha. *Anales AFA* **28**, 99-105 (2018).
- [7] M. Stipčević, B. G. Christensen, P. G. Kwiat y D. J. Gauthier. Advanced active quenching circuit for ultra-fast quantum cryptography. *Opt. Express* **25**, 21861-21876 (2017).
- [8] Y. Bromberg, Y. Lahini, E. Small e Y. Silberberg. Hanbury-Brown and Twiss Interferometry with Interacting Photons. *Nat. Photonics* **4** (ago. de 2010).
- [9] D. Branning y M. Beck. *An FPGA-based module for multiphoton coincidence counting* en *Advanced Photon Counting Techniques VI* **8375** (2012), 97-106.
- [10] B. K. Park, Y.-S. Kim, O. Kwon, S.-W. Han y S. Moon. High-performance reconfigurable coincidence counting unit based on a field programmable gate array. *Appl. Opt.* **54**, 4727-4731 (2015).
- [11] D. Branning, S. Bhandari y M. Beck. Low-cost coincidence-counting electronics for undergraduate quantum optics. *Am. J. Phys.* **77**, 667-670 (2009).
- [12] M. Bolaños. *Photon Coincidence Counter* <https://github.com/bmatiasruben/Deteccion-y-conteo-de-fotones>. 2020.
- [13] Australian National University. *ANU QRNG* <https://qrng.anu.edu.au/>.