

Optimización del hardware para medición de radiación utilizando sensores de imagen CMOS

Hardware optimization for radiation measurement using CMOS image sensors

Gaspar Domé^{*†}, M. C. Cebedio^{*†1}, M. Antonelli^{*†‡2}, J. Castiñeira Moreira^{*†‡3}, L. 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)

Argentina

{¹celestecebedio,²maxanto,³casti,⁴ldemicco}@fi.mdp.edu.ar

Received: 2025-12-13; Accepted: 2026-04-09

Resumen—Este artículo presenta el diseño, implementación y optimización en hardware de un sistema de detección de radiación basado en un transductor optoelectrónico de imagen CMOS comercial y una FPGA de ultra bajo consumo, orientado a aplicaciones espaciales con recursos computacionales y energéticos limitados. El trabajo aborda la implementación eficiente de un algoritmo de detección de eventos, previamente validado en software, en una FPGA Lattice iCE40HX4K. Para ello, se diseñó la arquitectura del sistema mediante un enfoque centrado en memoria, sustituyendo los métodos basados en registros, lo que permitió integrar funcionalidades como detección de eventos, filtrado de píxeles defectuosos, conteo y generación de histogramas, operando a una tasa de un cuadro por segundo. La validación experimental se realizó empleando secuencias reales de imágenes irradiadas, adquiridas previamente con el mismo transductor optoelectrónico. Los histogramas generados por la FPGA presentan una desviación inferior al 3 % respecto a un modelo de referencia implementado en MATLAB. Como resultado, la arquitectura optimizada logra una reducción superior al 90 % en el uso de recursos lógicos, manteniendo la funcionalidad original.

Palabras clave: radiación ionizante; FPGA, transductor optoelectrónico; algoritmo de detección; satélite.

Abstract—This article presents the design, implementation, and hardware optimization of a radiation detection system based on a commercial CMOS optoelectronic imaging transducer and an ultra-low-power FPGA, geared towards space applications with limited computational and energy resources. The work addresses the efficient implementation of an event detection algorithm, previously validated in software, on a Lattice iCE40HX4K FPGA. To achieve this, the system architecture was designed using a memory-centric approach, replacing register-based methods. This allowed for the integration of functionalities such as event detection, defective pixel filtering, counting, and histogram generation, operating at a rate of one frame per second. Experimental validation was performed using real sequences of irradiated images, previously acquired with the same optoelectronic transducer. The histograms generated by the FPGA show a deviation of less than 3 % compared to a reference model implemented in MATLAB. As a result, the optimized architecture achieves a reduction of over 90 % in the use of logical resources, while maintaining the original

functionality.

Keywords: ionizing radiation; FPGA; CMOS sensor; detection algorithm; satellite.

I. INTRODUCCIÓN

La detección de radiación ionizante en entornos espaciales es un desafío para la protección de los componentes electrónicos de satélites y otros dispositivos. Los componentes electrónicos en el espacio son altamente susceptibles a la radiación ionizante debido a la reducción drástica de la atenuación que provee la atmósfera a medida que se incrementa la distancia a la superficie de la Tierra.

Los efectos de la radiación provocan la degradación de las propiedades eléctricas de los dispositivos semiconductores, cambios en el rendimiento del sistema, errores suaves (*soft errors*) y *latch-up* [1], que alteran el comportamiento de los circuitos, provocando fallos funcionales y reduciendo la vida útil del hardware. Existen diversas tecnologías de detección, como detectores de semiconductores, contadores Geiger y cámaras de ionización, pero presentan limitaciones en términos de costo, tamaño, peso, consumo de energía y durabilidad [2].

Los efectos de la radiación sobre dispositivos CMOS pueden clasificarse en dos categorías principales: efectos puntuales, como los *Single Event Upsets* (SEUs), y la degradación acumulativa, como la dosis total ionizante (*Total Ionizing Dose* o TID). Los SEUs alteran transitoriamente el estado lógico de los circuitos, y suelen mitigarse mediante técnicas de redundancia o corrección de errores. En cambio, la degradación acumulativa representa un riesgo mayor, ya que puede provocar cambios permanentes en las características eléctricas de los transistores. Por ejemplo, la acumulación de carga en las capas de óxido puede generar un aumento en la corriente de fuga o una pérdida de control en la compuerta, lo que afecta la funcionalidad del dispositivo. A largo plazo, estas alteraciones pueden causar fallos irreversibles [3].

En este contexto, contar con un sistema de detección que,

no solo registre eventos ionizantes, sino que también permita caracterizarlos y analizar su impacto, se vuelve fundamental. Este tipo de herramienta puede ser útil tanto en tareas de validación en tierra como en aplicaciones en órbita, donde se requieren soluciones confiables, de bajo consumo y bajo costo, integrables en plataformas de escala reducida u otros dispositivos científicos.

En [4] se estudia la viabilidad de utilizar el transductor optoelectrónico óptico CMOS que utilizamos en este trabajo para la detección de radiación ionizante. Una propuesta de un algoritmo capaz de detectar y clasificar eventos se presenta en [5], allí el consumo de recursos de hardware es muy elevado y depende significativamente de la síntesis de alto nivel (HLS). El presente trabajo aborda la optimización del procesamiento y el diseño de un sistema más eficiente, implementado sobre una FPGA de bajo costo. El objetivo principal es desarrollar un sistema de detección de que optimice el uso de recursos disponibles sin comprometer la calidad de la medición. A diferencia de otros enfoques, que recurren a componentes especializados [6], este diseño aprovecha transductor optoelectrónico CMOS de producción masiva [7] y las ventajas de las FPGA de Lattice, ampliamente empleadas en aplicaciones aeroespaciales [8].

II. DESCRIPCIÓN DEL SISTEMA

El principio de funcionamiento se basa en la interacción de la radiación con el transductor optoelectrónico: cuando una partícula cargada impacta sobre la superficie del CMOS, deposita una carga que se manifiesta como una región con mayor brillo respecto a los píxeles circundantes.

La solución propuesta basa su arquitectura en un transductor optoelectrónico óptico CMOS MT9M001C12STM y una FPGA iCE40HX4K. La FPGA cumple un doble rol: realiza las tareas de control y procesamiento de datos, y gestiona la comunicación con la plataforma satelital.

II-A. Dimensionamiento de la fuente de alimentación y estimación de potencia

El dimensionamiento de la fuente de alimentación se realizó a partir del análisis de las hojas de datos del FPGA iCE40 LP/HX [9] y del sensor de imagen MT9M001 [10]. El sensor presenta un consumo típico de 363mW a 3.3V, correspondiente a la suma de sus alimentaciones digital, analógica y de píxeles (24mA, 85mA y 10mA respectivamente). Para una fuente de 1W se garantiza un margen adecuado que cubre los picos de corriente durante el arranque y la operación normal.

La contribución del FPGA se estimó considerando los 20 pines de E/S que conforman la interfaz con el sensor, los cuales operan a 48MHz. La potencia dinámica disipada por cada pin de salida se calcula como:

$$P_{\text{pin}} = \frac{1}{2} C_L V^2 f$$

donde C_L es la capacitancia de carga, V es la tensión de alimentación del banco de E/S y f es la frecuencia de conmutación. Asumiendo una carga total de 15pF por pin (capacitancia de entrada del sensor más parasitismos de la

placa) y operación con LVCMOS 2.5V ($V_{\text{CCIO}} = 2.62V$ máximo), la potencia dinámica por pin resulta:

$$P_{\text{pin}} = 0,5 \times 15 \text{ pF} \times (2,62 \text{ V})^2 \times 48 \text{ MHz} \approx 2,47 \text{ mW}$$

Para 20 pines activos, la potencia dinámica total de E/S es de aproximadamente 49.4 mW. Sumando la potencia del núcleo del FPGA, estimada entre 40 mW y 80 mW según la actividad lógica, el consumo total del FPGA se sitúa en el rango de 90 a 130 mW. De esta forma, la potencia total del sistema, incluyendo el sensor, se estima entre 450 mW y 500 mW, lo que valida la selección de una fuente de 1 W con holgura suficiente.

Para reducir el consumo energético del sistema, se proponen dos estrategias principales. La primera consiste en reemplazar la FPGA de la familia HX por un dispositivo de la familia LP (ultra low power), que reduce la corriente estática del núcleo de $1140\mu A$ a $250\mu A$, manteniendo la capacidad de operar a 48MHz requerida por el sensor. La segunda estrategia implementa un modo de captura intermitente, en el cual el sensor MT9M001 se mantiene en standby o con sus alimentaciones completamente desconectadas entre frames, activándose únicamente durante los intervalos necesarios para la adquisición de imágenes. En operación continua, el sistema presenta un consumo estimado de 493m W (130mW para la FPGA y 363mW para el sensor). Con las estrategias propuestas, este valor se reduce a aproximadamente 60mW (30mW para cada subsistema), lo que representa una disminución del 88% y habilita su uso en aplicaciones alimentadas por batería.

II-B. Configuración del transductor optoelectrónico

El sistema cuenta con un bloque inicial encargado de configurar los registros del transductor optoelectrónico CMOS mediante una interfaz de comunicación serie similar a I²C, adaptada a un enlace punto a punto. Dado que únicamente se requiere escribir registros durante esta etapa, no se implementa la funcionalidad de lectura. La simplificación del protocolo hace que la inclusión de un ID de dispositivo sea innecesaria, ya que la primera información transmitida corresponde directamente a una instrucción de lectura o escritura. La lógica de escritura de registros del transductor optoelectrónico se resume de la siguiente manera:

- Enviar un byte específico de escritura (xBA).
- Enviar el byte correspondiente al registro que se desea escribir.
- Enviar uno o dos bytes según el tamaño de dato requerido por el registro.
- Si se desea escribir un registro contiguo al anterior, se pueden enviar directamente los bytes de datos adicionales, ya que el transductor optoelectrónico incrementa internamente la dirección del registro.
- Para escribir un registro no contiguo, se envía una señal de stop y se inicia nuevamente el proceso de escritura.

II-C. Bloque de adquisición

El Bloque de Adquisición es responsable de recibir los datos provenientes del transductor optoelectrónico CMOS y

prepararlos para el procesamiento en la FPGA. El transductor optoelectrónico entrega las imágenes a través de varias señales:

- **Frame Valid:** Señala que la imagen no se encuentra en zonas de blanqueo vertical.
- **Line Valid:** Señala que la línea está activa.
- **PixCLK:** Reloj que sincroniza los datos.
- **Trigger:** Señal de inicio de adquisición.
- **d9–d2:** Bus de datos de 8 bits por pixel indicando el brillo.

La salida digital del transductor optoelectrónico posee 10 bits de resolución, los datos se truncan a 8 bits descartando los dos bits menos significativos, para simplificar el diseño, reducir el uso de memoria y realizar un filtrado previo.

Luego, para el manejo de estos datos en la FPGA:

- La FPGA recibe un reloj externo de 12 MHz que se lleva a 50.5 MHz mediante un PLL, lo que permite disponer de más ciclos de reloj para procesar los datos en tiempo real.
- Se genera un reloj de 2 MHz para el transductor optoelectrónico.
- Para gestionar el cruce de dominios de reloj entre el *pixel clock* del transductor optoelectrónico (2 MHz) y el reloj de procesamiento de la FPGA (50.5 MHz), se implementó un esquema de sincronización basado en registros y señales de validación para prevenir problemas de metastabilidad.

II-D. Algoritmo y optimización de memoria

Se optimizó el algoritmo propuesto en [5] para detectar eventos de interés y descartar el fondo de la imagen (Fig. 1).



Fig. 1: Ejemplo de detección de eventos de radiación (traza de partículas) sobre la imagen capturada [5].

El procesamiento se centra en la identificación de eventos (motas o puntos) compuestos por grupos de píxeles. Para cada evento, se suma la intensidad de todos los píxeles que lo constituyen, y con estos valores se genera un histograma. Dado que existe una correlación entre la carga depositada por una partícula ionizante y la respuesta del transductor [4], [7],

dicha suma de intensidades está relacionada con la energía de cada evento.

Al concluir la detección, el sistema asigna la intensidad de cada evento al bin correspondiente. Para evitar falsos positivos generados por ruido térmico o píxeles defectuosos, se descartan automáticamente los eventos de un solo píxel, obteniendo así una representación precisa de la distribución energética de la radiación incidente.

II-E. Optimización de arquitectura y análisis de casos límite

El sistema desarrollado originalmente en [5] empleó síntesis de alto nivel (HLS) sin restricciones estrictas de hardware, resultando en un uso ineficiente de las unidades lógicas. Aproximadamente el 80 % de los recursos se destinaron a registros (Flip-Flops) para implementar memorias internas. Esta arquitectura permitía modificar simultáneamente múltiples valores del vector de entrada en un único ciclo de reloj, facilitando una re-etiquetación global de píxeles cuando se detectaba la convergencia de dos eventos.

Al identificar estas limitaciones, en este trabajo se propone un enfoque centrado en bloques de memoria BRAM para reducir el uso intensivo de registros. Sin embargo, las memorias BRAM poseen restricciones de tiempo de acceso que impiden actualizar múltiples posiciones de memoria dentro del breve intervalo entre la llegada de píxeles consecutivos. Esta limitación física hace impracticable la re-etiquetación global en tiempo real bajo las mismas condiciones de operación.

El principal desafío de este cambio arquitectónico surge ante eventos con geometrías complejas (e.g., formas en “U” o “V” invertida). En estos casos, un único impacto físico es detectado inicialmente como dos eventos independientes que convergen en una línea posterior. Mientras que el diseño previo resolvía este conflicto unificando las etiquetas modificando los registros en pocos ciclos de reloj, el sistema optimizado no permite la re-etiquetación global debido a los tiempos de acceso de la BRAM.

Para evaluar el impacto de esta simplificación, se utilizan “imágenes de tortura” diseñadas para maximizar estos conflictos de convergencia. En estos escenarios límite, el algoritmo presenta fragmentación de eventos (un único impacto se contabiliza como múltiples eventos de menor intensidad), incrementando el error respecto al diseño original.

Sin embargo, las pruebas con datos reales de radiación demuestran que la ocurrencia de estas geometrías es despreciable en condiciones de operación normales. Por lo tanto, la reducción del 92 % en recursos lógicos compensa la pérdida de precisión en casos límite.

II-F. Mapeo de memoria

Una vez se demuestra la posibilidad de implementar el algoritmo en la EDU-CIAA-FPGA utilizando menos unidades lógicas de las disponibles, se evalúa si la memoria de la plataforma es suficiente para el funcionamiento del sistema.

Se propone el uso de tres bloques de memoria BRAM para el procesamiento de datos en tiempo real. El primero

consiste en una memoria circular de entrada que actúa como un buffer de línea (line buffer); esta almacena los índices de los eventos detectados en la línea de píxeles que está siendo barrida actualmente por el transductor. Dado que el transductor optoelectrónico entrega la información de manera secuencial (línea por línea), esta estructura permite al algoritmo comparar los píxeles de la línea actual con los de la anterior para determinar la conectividad del evento. Las otras dos memorias se utilizan para registrar, de forma acumulativa, la intensidad total y la cantidad de píxeles asociados a cada índice de evento identificado.

Para dimensionar las memorias, se procesaron secuencias de video de irradiación real suministradas por el Laboratorio de Bajas Temperaturas del Instituto Balseiro. Mediante la caracterización estadística de estos fotogramas, se extrajeron tres parámetros máximos: el tamaño del evento (cantidad de píxeles), la intensidad acumulada por evento y la cantidad de eventos detectados simultáneamente.

Con estos parámetros, se definió la **profundidad** (cantidad de direcciones) y su **ancho de palabra** (cantidad de bits) de los tres bloques de memoria BRAM:

- **Memoria circular de entrada (Line Buffer):** Almacena los índices de los eventos de la línea actual. Su profundidad es igual al ancho en píxeles del transductor optoelectrónico con tres píxeles adicionales por el funcionamiento del algoritmo, mientras que su ancho de palabra está definido por la cantidad de bits necesarios para direccionar el número máximo de eventos simultáneos.
- **Memoria de cantidad de píxeles:** Su profundidad corresponde a la cantidad máxima de eventos simultáneos (una dirección por evento). Su ancho de palabra se calcula en función de los bits requeridos para almacenar el tamaño máximo de píxeles registrados para un evento, extraído del análisis estadístico.
- **Memoria de intensidad de eventos:** Su profundidad también corresponde a la cantidad máxima de eventos simultáneos. Su ancho de palabra está definido por los bits necesarios para representar la intensidad total máxima acumulada por un evento, extraído del análisis estadístico.

Para garantizar la robustez del sistema frente a escenarios fuera del promedio, se añadió un bit adicional tanto al bus de direcciones como al bus de datos de cada memoria, previniendo así errores por desbordamiento.

II-G. Histograma

En esta etapa se define el histograma considerando los requerimientos del sistema y las características del transductor optoelectrónico CMOS. Se utiliza un histograma de 32 bins de ancho variable, calculado según la resolución de intensidad presentada en (1).

$$\text{Ancho del bin} = \frac{2^N}{32} \quad (1)$$

Cada píxel tiene una resolución de 10 bits, recortada a 8 bits durante la adquisición. Dado que la suma de las intensidades de los píxeles que componen un evento no supera los 2^{14} bits —según se verificó con los datos procesados—, se define $N = 14$ bits para abarcar el rango dinámico de la intensidad acumulada por evento. De esta manera, el ancho máximo de cada bin resulta:

$$\text{Ancho de bin máximo} = \frac{2^{14}}{32} = 512 \quad (2)$$

El histograma se genera a partir de los eventos detectados por el algoritmo diseñado en la etapa anterior y únicamente se procesan aquellos que contienen más de un píxel, descartando píxeles aislados que corresponden a ruido o píxeles muertos. El valor de intensidad total de cada evento se asigna al bin correspondiente, y el resto de la memoria BRAM se utiliza para almacenar los histogramas de manera circular, esperando la solicitud del satélite para la descarga de datos. Para obtener un histograma representativo, se combinan 60 imágenes capturadas durante un minuto en un solo histograma, que luego se guarda en memoria.

II-H. Controlador

Con todos los bloques: capturador, algoritmo, banco de memoria y generador de histogramas se diseña un controlador que coordine todas las máquinas de estado y maneje la comunicación con el satélite, esperando un comando para reiniciar si fuese necesario y otro para descargar toda la información.

III. RESULTADOS

En esta sección se presentan los resultados obtenidos del funcionamiento general del sistema y luego, aquellos relevantes a cada sección.

III-A. Configuración del transductor optoelectrónico

La Figura 2 muestra el protocolo de comunicación funcionando en un osciloscopio. La señal en rojo es el clock del sistema con un frame de inicio, y la señal en amarillo muestra la señal de configuración del CMOS.



Fig. 2: Señales del protocolo de configuración del transductor optoelectrónico CMOS observadas en el osciloscopio.

III-B. Bloque captador

Para probar el correcto funcionamiento del bloque captador se reciben los datos sin procesar en una PC y luego se reconstruyen. El transductor optoelectrónico genera 8 bits en cada ciclo de reloj que se transmite a través de un bus paralelo de 8 bits, por lo que la tasa de datos resulta como se indica en 3:

$$bps = 8 \cdot f = 8 \cdot 2 \text{ MHz} = 16 \text{ Mbps} \quad (3)$$

Como el sistema utiliza un enlace de comunicación serie entre la FPGA y la PC, la tasa de datos generada resulta demasiado alta para ser transmitida directamente. Para resolver este problema, se emplea la placa de desarrollo DE2-115, que incorpora una memoria SRAM de 2 MB. Esta memoria permite almacenar temporalmente los píxeles a la velocidad necesaria durante la adquisición, y luego enviarlos a la PC a una tasa reducida, compatible con la interfaz serie.

III-C. Optimización del algoritmo

En esta etapa se registran los resultados obtenidos con la alternativa de eliminar los casos problemáticos y se analizan sus efectos sobre los resultados. Para evaluar el desempeño del algoritmo simplificado frente al diseño original, se utilizó un conjunto de imágenes de tortura: imágenes sintéticas generadas específicamente para probar casos límite en la etapa de etiquetado, donde se incluyen configuraciones de píxeles mucho más complejas que las que se presentan en mediciones reales.

El algoritmo va recibiendo los píxeles y asigna a cada uno una etiqueta numérica según el evento al que pertenece. Así, todos los píxeles del primer evento reciben la etiqueta "1", los del segundo la etiqueta "2", y así sucesivamente. Para comparar visualmente la exactitud del algoritmo original frente al simplificado, se asigna un color diferente a cada etiqueta. De esta forma, en una imagen correctamente procesada, idealmente cada grupo de píxeles (evento) debería tener un único color.

La Fig. 3 muestra los resultados obtenidos al aplicar ambos algoritmos sobre las mismas imágenes de tortura. En ella se evalúa cómo se verían las imágenes cuando el algoritmo considera todos los casos (diseño original) frente al algoritmo simplificado. Puede observarse que, en las imágenes de tortura, el nuevo algoritmo presenta una mayor cantidad de fallas en comparación con el diseño previo.

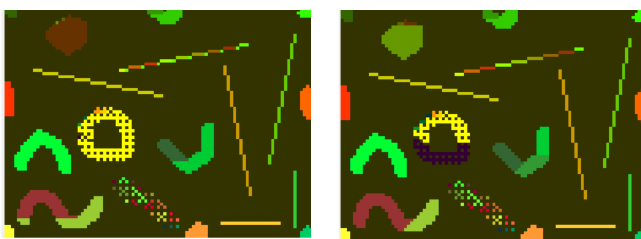


Fig. 3: Comparación de algoritmos sobre imágenes sintéticas "de tortura". Cada color se corresponde con una etiqueta asignada por el algoritmo.

Sin embargo, bajo condiciones reales, los eventos a procesar son mucho más simples que las imágenes sintéticas de tortura. Utilizando imágenes obtenidas a partir de fuentes reales de radiación, no se observan diferencias significativas en el rendimiento. Por lo tanto, se considera que el nuevo algoritmo es válido para aplicaciones prácticas. Como ejemplo, se presenta en la Fig. 4, el resultado de aplicar el algoritmo optimizado sobre datos reales, en esta imagen, cada evento tiene forma circular o ligeramente ovalada, entonces el algoritmo realiza el etiquetado correctamente y no aparecen eventos con más de un color. Por otro lado, algo esencial y

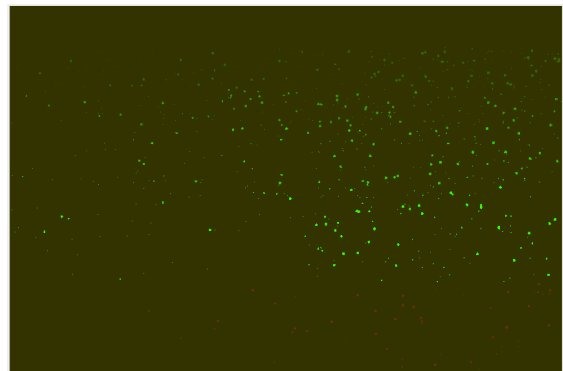


Fig. 4: Resultado del algoritmo optimizado aplicado a datos reales de radiación. Los eventos reales son simples de procesar, por lo que el algoritmo los etiqueta correctamente, resultando en un color uniforme para cada uno de ellos.

parte del objetivo del trabajo es reducir el hardware asociado a la implementación del algoritmo. En el Cuadro I, se observan los recursos utilizados en la implementación de los dos algoritmos y la reducción que genera el nuevo enfoque.

| Componente | Sistema previo | Sistema actual | Reducción % |
|-----------------------|----------------|----------------|-------------|
| Unidades Lógicas (LU) | 24,792 | 1,945 | 92.16 |
| Flip-Flops (FF) | 27,043 | 1,024 | 96.22 |

Cuadro I: Comparación del uso de recursos lógicos entre el sistema previo y el sistema optimizado.

III-D. Mapeo de memoria

El resultado de dimensionar las memorias se observa en el Cuadro II. La memoria total requerida únicamente para el algoritmo es de aproximadamente 8 kilobytes, lo que se encuentra por debajo de los 10 kilobytes disponibles en la FPGA iCE40HX4K.

| Memoria | Tamaño |
|--------------|-----------------------|
| Entrada | 2048 × 12 bits ≈ 3 kB |
| Intensidad | 2048 × 14 bits ≈ 3 kB |
| Cantidad | 2048 × 6 bits ≈ 2 kB |
| Total | 8 kB |

Cuadro II: Dimensionamiento de las memorias utilizadas por el algoritmo de procesamiento de eventos.

III-E. Histograma

La etapa de validación del histograma, representa una validación del sistema completo, debido a que todos los bloques deben contribuir en la realización del mismo. Para

verificar el correcto funcionamiento del sistema, se diseña un banco de pruebas, presentado en la figura 5.

Este banco de pruebas incorpora una memoria donde se almacena la imagen sin procesar capturada por el sensor. Su función es permitir la transferencia a la computadora tanto del histograma generado por la FPGA a partir de los datos, como de la imagen sin procesar. Una vez en la computadora, se genera un segundo histograma a partir de esa imagen sin procesar y se compara con el recibido desde la FPGA. La validación se basa en la coincidencia entre ambos histogramas, además, la PC genera una imagen con los eventos coloreados según el número de etiqueta del evento al cual pertenecen, como una forma de validar que la imagen capturada por el sensor es correcta.

En esta etapa se utiliza la plataforma de desarrollo DE2-115, principalmente debido a que cuenta con suficiente capacidad de memoria para almacenar una imagen completa capturada por el transductor optoelectrónico. Lo cual es fundamental, ya que permite garantizar que los datos enviados a la computadora para su procesamiento corresponden exactamente a la misma imagen procesada en la FPGA, condición necesaria para que la comparación de histogramas tenga validez.

Se desarrollaron bloques específicos para el manejo de la memoria, también se implementó un bloque de coordinación que sincroniza el almacenamiento de la imagen y su posterior transmisión a la computadora.

Adicionalmente, se implementa un programa en MATLAB encargado de recibir los datos provenientes de la FPGA, reconstruir la imagen, aplicar el mismo algoritmo utilizado en la FPGA respetando las limitaciones de resolución en bits del hardware, y generar el histograma de referencia. Todo este proceso asegura una comparación válida entre el histograma generado por la FPGA y el histograma de referencia obtenido en la computadora.

El histograma se construye mediante el barrido secuencial de las memorias de intensidad y de cantidad de píxeles, las cuales poseen una correspondencia biunívoca entre sus direcciones. Cada evento es clasificado en bins de ancho fijo según la Ec. (1), donde el eje de las abscisas representa el nivel de “intensidad acumulada” y el eje de las ordenadas la frecuencia de ocurrencia. Por ejemplo, para un ancho de 512 ($2^{14}/32$), el primer bin contabiliza los eventos con intensidades en el rango $[0, 512)$, el segundo entre $[512, 1024)$ y así sucesivamente.

IV. DISEÑO PCB

Se diseña una placa basada en las recomendaciones del grupo LabOSat, siguiendo buenas prácticas de diseño [11] y adaptada a los requisitos del sistema. Si bien se toman detalles de desacople, selección de componentes periféricos y criterios de ruteo de la EDU-CIAA-FPGA desarrollada en la UTN de Haedo, el diseño no replica dicha placa, sino que se realiza un desarrollo nuevo ajustado a las necesidades específicas del proyecto. En particular, se elimina el programador reemplazándolo con un PCB externo que cumple

la misma función, se quitan las borneras de prototipado y se modifica por completo el sistema de alimentación para adaptarlo a las tensiones disponibles en el satélite.

IV-A. Transductor optoelectrónico

Los elementos adicionales requeridos para el funcionamiento del transductor optoelectrónico se determinan a partir de las especificaciones del datasheet [9] y del trabajo realizado en [5]. El diseño incorpora el transductor optoelectrónico, resistencias de pull-up necesarias para la interfaz de comunicación serie, y capacitores de desacople para garantizar la estabilidad de la alimentación.

IV-B. FPGA

Para el correcto funcionamiento de la FPGA, se consideran recomendaciones provistas en notas de aplicación de Lattice [10] y en el desarrollo de la EDU-CIAA-FPGA [12]. Se toma como referencia la selección de componentes periféricos, pero el diseño se adapta a las necesidades del sistema, sin replicar la placa original.

Se decide separar el programador de la placa final, desarrollando un módulo independiente que pueda ser reutilizado. Además, se eliminan las borneras de prototipado, dado que no son necesarias para el funcionamiento del sistema a bordo.

IV-C. Alimentación

Se estima que el satélite entrega una tensión entre 9 y 12 volts, mientras que el sistema requiere tensiones de 3.3 V y 1.2 V. De acuerdo a los datos provistos en los datasheets y las estimaciones realizadas, el consumo máximo esperado es de aproximadamente 300 mA a 3.3V. Este valor es consistente con la estimación detallada presentada en la Sección II-A, donde se considera el consumo conjunto del transductor optoelectrónico, la FPGA y las pérdidas de conversión de la fuente de alimentación.

Para mejorar la eficiencia, se descartan reguladores lineales directos y se opta por una fuente switching DC-DC para generar 3.3V, seguida de un regulador lineal que produce 1.2V. Además, se incorporan protecciones contra inversión de polaridad, fusibles de sobrecorriente y protección contra sobretensión.

La alimentación se organiza en tres etapas: primero, un regulador switching LM2576 genera 3.3V; segundo, un circuito de protección contra sobrevoltaje basado en un diodo Zener y dos transistores; y finalmente, un regulador lineal LDL1117S12R produce los 1.2V necesarios para el sistema.

IV-D. PCB terminada

En las Fig. 6 se presenta el PCB del sensor que incorpora la FPGA, el transductor optoelectrónico y la fuente de alimentación, sus medidas son 10cm por 6cm. En la Fig. 7 se presenta el PCB del programador, que incluye un conector USB tipo C, un conversor USB-SPI y la bornera de conexión compatible con el sensor, sus medidas son 6cm por 2cm.

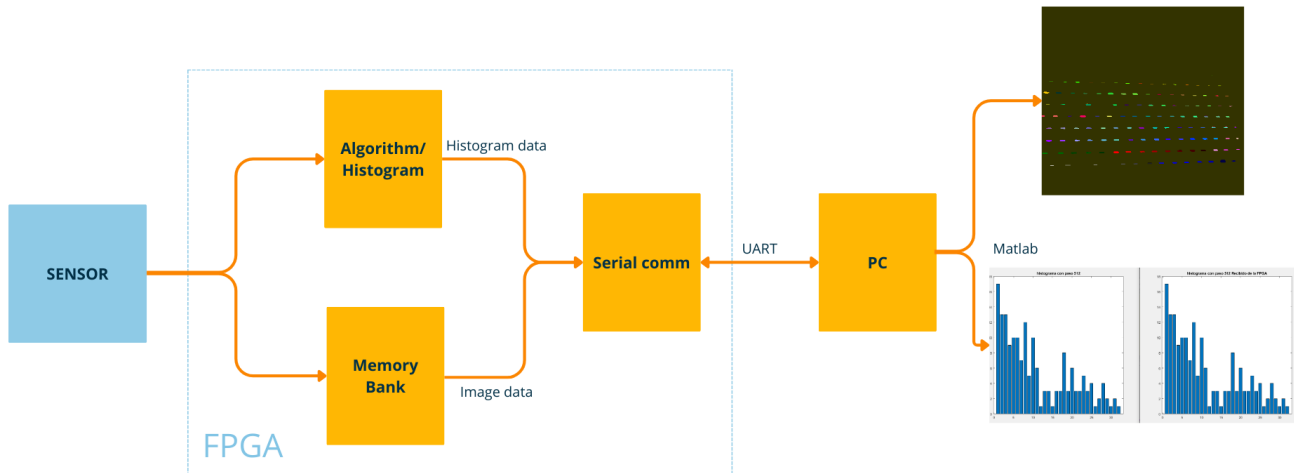


Fig. 5: Banco de pruebas utilizado para la validación del histograma generado por la FPGA.

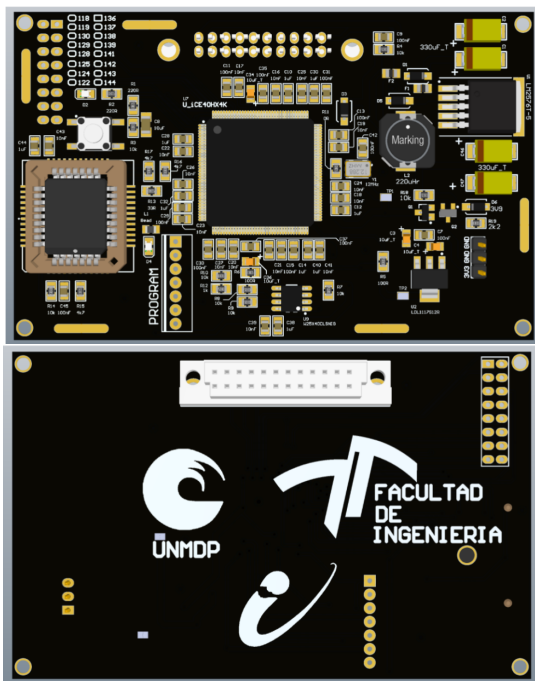


Fig. 6: Vistas frontal y posterior del PCB final que integra el transductor optoelectrónico CMOS, la FPGA y la etapa de alimentación.

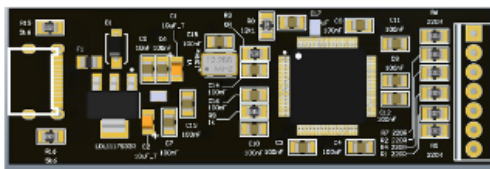


Fig. 7: PCB del programador externo utilizado para la configuración de la FPGA.

V. RESULTADOS

En esta sección se evalúa el desempeño del sistema optimizado en términos de ocupación de recursos de hardware, consumo y fidelidad en la clasificación de eventos.

V-A. Eficiencia en el uso de recursos

El rediseño del algoritmo utilizando el lenguaje VHDL y priorizando el uso de bloques de memoria BRAM, en lugar de una arquitectura basada puramente en registros, permitió una reducción drástica en la ocupación de la FPGA. En el Cuadro I se detalla que se logró una disminución del 92.16 % en unidades lógicas (LUTs) y del 96.22 % en Flip-Flops (FFs). Estos resultados permiten que el sistema completo utilice menos del 60 % de la capacidad de la FPGA iCE40HX4K, posibilitando la futura integración de módulos adicionales.

V-B. Validación funcional

El sistema se validó mediante la comparación de los resultados procesados en la FPGA frente a un modelo de referencia desarrollado en MATLAB. Para esta evaluación se utilizaron “imágenes de tortura” que contienen eventos con geometrías complejas, diseñadas específicamente para estresar la lógica de clasificación y analizar los límites del algoritmo.

Si bien el algoritmo optimizado presenta variaciones en la segmentación de estos casos límite de geometría no natural, su desempeño en imágenes con trazas de radiación real es consistente con el modelo de referencia. El hardware procesa los datos a una tasa de un cuadro por segundo.

Bajo condiciones normales de operación, utilizando imágenes obtenidas a partir de fuentes reales de radiación, no se observan diferencias significativas en el rendimiento respecto al modelo de referencia. En la Fig. 4 se presenta el resultado de aplicar el algoritmo optimizado sobre estos datos. Cada grupo de píxeles coloreados representa la traza dejada por la deposición de energía de una partícula ionizante individual sobre el sensor CMOS MT9M001. La visualización confirma la capacidad de la arquitectura propuesta para aislar eventos de radiación del ruido de fondo y clasificarlos correctamente en tiempo real.

VI. CONCLUSIONES

Se desarrolló e implementó un sistema de detección de radiación ionizante utilizando una FPGA de bajo consumo y reducidas unidades lógicas junto a un transductor optoelectrónico de imagen. La optimización permitió una reducción superior al 90 % en el uso de recursos lógicos respecto a implementaciones previas, manteniendo la capacidad de generar histogramas de intensidad de eventos.

La validación mediante modelos confirma que el sistema es capaz de caracterizar eventos de radiación en tiempo real. Este diseño ofrece una solución de bajo costo y reducida utilización de área, adecuada para la monitorización de dosis en satélites de pequeña escala. Como trabajo futuro, se planea la fabricación de la versión final del PCB y la realización de pruebas de funcionamiento en vuelo para evaluar el desempeño del sistema en condiciones orbitales.

DECLARACIÓN DE DISPONIBILIDAD DE DATOS

Los datos que respaldan los resultados de este estudio están disponibles a partir del autor de correspondencia previa solicitud razonable.

CREDIT DECLARACIÓN DE CONTRIBUCIÓN DE AUTORÍA

Gaspar Domé: Software; investigación; Validación. **M. Celeste Cebedio:** redacción – borrador original. **Maximiliano Antonelli:** investigación; Software; visualización; redacción. **Jorge Castiñeira Moreira:** supervisión. **Luciana De Micco:** investigación; visualización; Software; redacción. **Raúl E. Lopresti:** supervisión.

REFERENCIAS

- [1] R. D. Schrimpf and D. M. Fleetwood, *Radiation Effects and Soft Errors in Integrated Circuits and Electronic Devices*. Singapore; New Jersey: World Scientific, 2004, doi: 10.1142/5607.
- [2] G. F. Knoll, *Radiation Detection and Measurement*, 3rd ed. Hoboken, NJ: John Wiley & Sons, 1999, chapter 5, 7, 11, 12 y 13.
- [3] A. H. Johnston, “Radiation damage of electronic and optoelectronic devices in space,” *4th International Workshop on Radiation Effects on Semiconductor Devices for Space Application*, 2000, accedido: abril 2025. [Online]. Available: https://nepp.nasa.gov/docuploads/D41D389D-04D4-4710-BBCFF24F4529B3B3/Dmg_Space-00.pdf
- [4] M. Pérez, J. Lipovetzky, M. Sofo Haro, I. Sidelnik, J. J. Blostein, F. Alcalde Bessia, and M. Gómez Berisso, “Particle detection and classification using commercial off the shelf cmos image sensors,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 827, pp. 171–180, 2016, doi: 10.1016/j.nima.2016.04.072.
- [5] M. Rolón and M. S. García, “Detector de radiación ionizante para aplicaciones satelitales,” Tesis de grado, Universidad Nacional de Mar del Plata (UNMDP), Mar del Plata, Buenos Aires, Argentina, 2021.
- [6] T. Berger, D. Matthiä, S. Burmeister, C. Zeitlin, R. Rios, N. Stoffle, N. A. Schwadron, H. E. Spence, D. M. Hassler, B. Ehresmann *et al.*, “Long term variations of galactic cosmic radiation on board the international space station, on the moon and on the surface of mars,” *Journal of Space Weather and Space Climate*, vol. 10, p. 34, 2020, doi: 10.1051/swsc/2020028.
- [7] J. Lipovetzky *et al.*, “Evaluation of commercial off-the-shelf cmos image sensors for total ionizing dose radiation detection,” *Radiation Physics and Chemistry*, vol. 177, p. 109156, 2020, doi: 10.1016/j.radphyschem.2020.109062.
- [8] M. Wirthlin, “High-reliability fpga-based systems: Space, defect tolerance, and fault injection,” *Proceedings of the IEEE*, vol. 103, no. 3, pp. 379–392, 2015.
- [9] Micron, “1/2-inch megapixel cmos digital image sensor,” Datasheet, 2006. [Online]. Available: https://github.com/Fresita-codificadora/Proyecto_Final_DOME/blob/main/datasheet/HOJA_DE_DATOS_IMAGER_mt9m001_USADO_CON_FPGA.pdf
- [10] Lattice Semiconductor Corporation, *ICE40 LP/HX Family Handbook*, 2017, accedido el 13 de abril de 2025. [Online]. Available: <https://www.latticesemi.com/~media/latticesemi/documents/handbook/s/ice40familyhandbook.pdf>
- [11] Texas Instruments, “Pcb design guidelines for reduced emi,” 2016. [Online]. Available: <https://www.ti.com/lit/an/szza009/szza009.pdf>
- [12] CIAA - Computadora Industrial Abierta Argentina, “Edu-ciaa-fpga - computadora industrial abierta argentina,” accedido el 13 de abril de 2025. [Online]. Available: <https://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=edu-ciaa:fpga:start>