Instrumentación de un arreglo de microbolómetros de 120x160

por

Julisa Verdejo Palacios

Tesis presentada en cumplimiento parcial de los requisitos para el grado de:

Maestría en Ciencias en la Especialidad de Electrónica

en

Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)

Octubre, 2023

Santa María de Tonantzintla, Puebla

Asesor:

Dr. Mario Moreno Moreno
Departamento de Electrónica INAOE

Agradecimientos

- Para mi familia, mis hermanos Efraín, Alejandro y mis padres Juana y Efrain, les agradezco con todo mi corazón su apoyo.
- A mis compañeros de laboratorio, Victor, Juan y Julio, gracias por ayudarme resolviendo mis dudas y por hacer el laboratorio un lugar de trabajo agradable.
- A mi novia Julisa, gracias por regañarme y hacerme ver mis virtudes y mis defectos y sobre empujarme a seguir adelante.
- A mi asesor Esteban, gracias por su infinita paciencia y apoyo, el trabajo que realizamos juntos representa el inicio de mi carrera profesional.

Índice general

Agradecimientos I

Índice general III

Índice de figuras V

Índice de tablas VII

Índice de códigos IX

Resumen XI

1. Introducción 1
- 1.1. Objetivos 1
  - 1.1.1. Objetivo general 1
  - 1.1.2. Objetivos específicos 2

2. Generadores de números aleatorios (RNGs) 3

3. Mapas caóticos 5

4. Implementación de TRNG híbrido 7

5. Resultados experimentales 9

6. Conclusiones 11

A. Códigos 13
- A.1. Códigos en C 13

Bibliografía 15

Índice de figuras

2.1. Jitter del reloj. 3

Índice de tablas

2.1. Parámetros recomendados para el conjunto de pruebas del NIST. . . . 4

Índice de códigos

Resumen

En esta tesis se diseñó e implementó un TRNG híbrido en la FPGA Xilinx Artix 7 xc7a35tcpg236-1 sobre la tarjeta de desarrollo Digilent Basys 3. Se utilizó un núcleo ERO-TRNG para generar una semilla de 64 bits que funciona como condición inicial para un mapa caótico bidimensional. Haciendo uso de la operación mod 256 se extraen 16 bits aleatorios por cada iteración del mapa.

En este trabajo se presenta toda la teoría necesaria para comprender los generadores de números aleatorios, así como su clasificación, fuentes de aleatoriedad, parámetros de evaluación pruebas estadísticas y arquitecturas de núcleos TRNG específicas para FPGA. Se estudian brevemente las características principales de los mapas caóticos como puntos fijos, estabilidad lineal, diagramas de bifurcación y diagramas de cobwebs. Después se expone la metodología de diseño para implementar en FPGA el mapa caótico bidimensional y el núcleo ERO-TRNG utilizando el lenguaje de descripción de hardware VHDL. El mapa caótico se implementó utilizando aritmética de punto fijo de 64 bits, 3 bits para la parte entera, 60 bits para la fraccionaria y un bit de signo y para comprobar su funcionamiento se empleó un simulador desarrollado lenguaje C. Posteriormente se analizó el dominio de atracción del mapa para diferentes parámetros con el fin de poder seleccionar un rango en que las condiciones iniciales produzcan caos, por ultimó se utilizaron multiplicadores de una sola constante para reducir el uso de recursos. Utilizando diversos elementos digitales básicos y el núcleo ERO-TRNG se diseñó un generador de semillas de 64 bits que alimenta a las condiciones iniciales del mapa caótico. Finalmente, las secuencias binarias obtenidas por el TRNG híbrido se mandaron a una computadora utilizando el protocolo de comunicación RS232 y se analizaron con las pruebas estadísticas NIST SP 800-22.

Capítulo 1

Introducción

Los números aleatorios se utilizan en muchos ámbitos de la vida cotidiana. Se utilizan para elegir quién gana la lotería, para determinar quién ataca primero en un partido de fútbol, para garantizar una partida justa en juegos de mesa y desempeñan un papel fundamental en la criptografía y la seguridad de la información. Para seleccionar al equipo atacante en un partido de fútbol, basta con lanzar una moneda. Sin embargo, para jugar a un juego de mesa se requieren más de dos valores aleatorios, por lo que se utiliza un dado. En cambio, la criptografía requiere algo más que tirar un dado para asegurar la protección de los datos en comunicaciones digitales o en transacciones bancarias. La seguridad de las comunicaciones es una parte fundamental de la vida moderna, en la que las personas envían correos electrónicos, realizan llamadas o envían mensajes a sus amigos y realiza transacciones en línea millones de veces al día. La sociedad confía que cada uno de estos procesos cotidianos sean seguros y confidenciales. La seguridad de las comunicaciones dependen de la capacidad de estos procesos para verificar la identidad de las personas que se comunican. La única forma de garantizar la seguridad es mediante la distribución de identidades privadas conocidas solo por el usuario, denominadas claves. Las claves privadas son números aleatorios únicos generados para cada usuario, que aseguran que personas malintencionadas no puedan suplantar a nadie y causar daño. La aleatoriedad de los números de las claves privadas es crucial para garantizar la seguridad de las conexiones. La capacidad de generar números aleatorios es, por tanto, una parte muy importante de la seguridad de los sistemas de comunicación.

1.1. Objetivos

1.1.1. Objetivo general

- Diseñar e implementar en FPGA un TRNG híbrido para la generación de secuencias muy largas.

1.1.2. Objetivos específicos

- Investigar el estado del arte de diferentes generadores de números aleatorios.
- Estudiar los diferentes tipos de generadores de números aleatorios y analizar sus características principales.
- Estudiar la teoría de los mapas caóticos y su utilidad en generadores de números aleatorios.
- Diseñar un generador de números aleatorios híbrido utilizando un TRNG como generador de semillas y un mapa caótico para realizar un postprocesamiento que mejore sus características estadísticas y comprobar estas utilizando las pruebas NIST.
- Implementar el TRNG híbrido en una FPGA.

Capítulo 2

Generadores de números aleatorios (RNGs)

En este capitulo se presentan los conceptos necesarios para comprender cómo funcionan las diferentes clases de generadores de números aleatorios, sus fuentes de aleatoriedad, sus principales características y cuales son factibles para implementación en FPGA.

El jitter del reloj en un sistema digital es una desviación del flanco de reloj real con respecto a un flanco de reloj ideal. Una señal de reloj ideal se define mediante la ecuación (2.1), donde $t(n)$ representa el tiempo del periodo n-ésimo de una señal de reloj y T es el periodo de una señal de reloj.

$$t(n) = n \cdot T \tag{2.1}$$

En la práctica, una señal de reloj real no alcanza siempre múltiplos enteros de su periodo ideal, sino que sus flancos fluctúan alrededor de este valor debido al jitter. Esta variación es causada por diversos fenómenos físicos, como el ruido térmico, el ruido de la fuente de alimentación y el ruido electromagnético ambiental, entre otros. La Figura 2.1 muestra cómo se ve una señal de reloj afectada por el jitter.

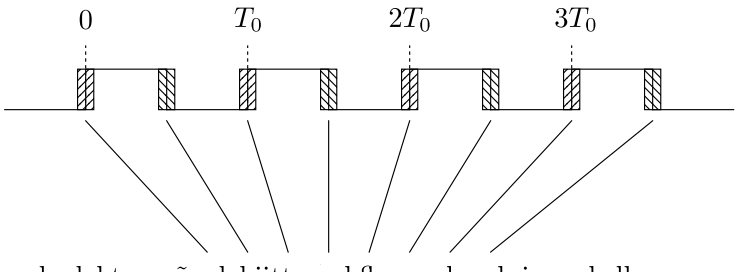

Figura 2.1: Jitter del reloj.

Los parámetros recomendados para configurar las pruebas NIST se muestran en la Tabla 2.1:

Tabla 2.1: Parámetros recomendados para el conjunto de pruebas del NIST.

Test	**Configuration item**	**Setting**
All tests	Bits per sequence	1000000
All test	Number of sequences (sample size)	1073
Frequency test within a block	Block length	20000
Non-overlapping template test	Template length	10
Overlapping template	Block length	10
Mauler's Universal Statistical test	Test block length L	7
Mauler's Universal Statistical test	Initialization steps	1280
Approximate entropy test	Block length	8
Linear complexity test	Block length	1000
Serial test	Block length	16

Capítulo 3

Mapas caóticos

En este capítulo se estudiarán los mapas caóticos utilizando como ejemplo el mapa logístico para entender cómo produce el caos y las diferentes técnicas para ver de manera cualitativa este fenómeno.

Capítulo 4

Implementación de TRNG híbrido

En este capítulo se describen los pasos para diseñar un mapa caótico y un ERO-TRNG en FPGA para generar bits aleatorios que servirán como semilla a las condiciones del mapa. Se analiza cómo seleccionar el tamaño de palabra de punto fijo utilizando un simulador en C, cómo mejorar el uso de recursos con multiplicadores de una constante y cómo seleccionar el rango valido de la semilla utilizando el dominio de atracción del mapa caótico.

Capítulo 5

Resultados experimentales

En este capítulo se presentan los resultados obtenidos de la implementación del TRNG híbrido, uso de recursos, velocidad de salida y pruebas estadísticas.

Capítulo 6

Conclusiones

- El núcleo ERO-TRNG a pesar de no ser el más rápido de los generadores que se pueden implementar dentro de las FPGA, tiene muy buena entropía, un modelo estocástico muy bien estudiado, una metodología de diseño bien estructurada y por lo mismo es fácil de implementarse no requiriendo intervención manual en la colocación de los componentes dentro del FPGA. Debido a lo anterior fue perfecto para generar semillas, una vez obtenida la semilla la velocidad del sistema ya no depende del ERO-TRNG lo que contrarresta su desventaja principal. Su repetibilidad en diferentes familias de FPGA, la seguridad que agrega al generador y la pequeña cantidad de recursos que requiere son las ventajas que presentó este núcleo.

- El mapa caótico bidimensional que se seleccionó para este trabajo es muy flexible, ya que se tiene 12 parámetros diferentes para configurarse, lo que se ve reflejado en diferentes atractores que pueden utilizarse para generar secuencias de bits aleatorias. Mientras las condiciones iniciales del atractor se encuentren dentro del dominio de atracción de este, podemos sembrar el mapa sin problemas.

- La implementación del TRNG híbrido utiliza un 88 % de los DPS y tan solo un 7 % de los LUTs de la FPGA, no obstante considerando que la FPGA utilizada es de bajos recursos, para FPGAs más grandes las cuales tienen de 4 a 5 recursos, este sistema no representa un gran consumo de área.

- El TRNG híbrido que se diseño en este trabajo pasó todas las pruebas NIST y el análisis estadístico demostró distribuciones uniformes, además debido a que en su núcleo se encuentra un TRNG que pasa todas las pruebas de la AIS20/31 la seguridad del sistema se ve garantizada mientras nadie tenga acceso a la semilla.

- La velocidad obtenida por el TRNG híbrido fue de 533.33 Mbit/s, es ligeramente superior a sistemas similares las cuales rondan los 400 Mbit/s.

Apéndice A

Códigos

A.1. Códigos en C

Bibliografía