

Trabajo Fin de Máster
Máster en Ingeniería Electrónica, Robótica y Au-
tomática

Aerial co-workers: a task planning approach
for multi-drone teams supporting inspection
operations

Autor: Álvaro Calvo Matos

Tutor: Jesús Capitán Fernandez

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021



Trabajo Fin de Máster
Máster en Ingeniería Electrónica, Robótica y Automática

Aerial co-workers: a task planning approach for multi-drone teams supporting inspection operations

Autor:

Álvaro Calvo Matos

Tutor:

Jesús Capitán Fernandez

Associate Professor

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021

Trabajo Fin de Máster: Aerial co-workers: a task planning approach for multi-drone teams
supporting inspection operations

Autor: Álvaro Calvo Matos
Tutor: Jesús Capitán Fernandez

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

Agradecimientos

A mi tutor, Hipólito, por guiarme en este proyecto, pero sobre todo, por darme la oportunidad de formar parte de su equipo desde tan temprano, otorgándome la plaza de alumno interno que me ha llevado a la realización de este TFG. A pesar de los momentos en los que ha escaseado el tiempo, ha sido una experiencia muy buena que reperitía sin dudarlo.

A María, por su ayuda cada vez que la he necesitado, a Luis, por guiarme en los inicios con la programación y solucionarme cada nuevo problema que me surgía, y a Dani, por las pequeñas ayudas que ha realizado.

A todos aquellos profesores que demostraban día a día la pasión por su trabajo y que hacían de aprender la mejor de las experiencias.

A mis maestros y maestras del Colegio Salesiano de Utrera, por su dedicación. Especialmente a mis profesores Elena y Fernando, por sus enseñanzas, pero sobre todo por su cariño, su comprensión y su paciencia conmigo. Y en particular a Eduardo, por todo lo que aprendí de informática y de física, pero sobre todo, por hablarme de los campus científicos que me han traído directamente hasta aquí.

A mis compañeros de clase, por acompañarme durante todo el camino.

A Damián, por su apoyo y amistad en todo momento durante este último año.

A mis amigos de Utrera, por tener paciencia y no haberme abandonado tras cuatro años desaparecido con mis estudios.

A mi familia, por su amor y apoyo incondicional.

A mi abuelo, que a pesar de no haber podido estar aquí, se ha ido dejándome la mejor herencia que se puede dejar, una buena educación.

Álvaro Calvo Matos
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Sevilla, 2020

Resumen

El diagnóstico de *Conmutaciones por evento singular o Single Event Upset (SEU)* es un problema abierto sobre el que apenas se han realizado investigaciones previas. En este trabajo perseguimos diseñar una nueva técnica de diagnóstico que permita localizar un SEU a partir de la información de la que se disponga.

Es común disponer únicamente de diccionarios de fallos incompletos, ya que, en circuitos grandes, el tiempo necesario para obtener un diccionario de fallos completo lo hace inviable. Vamos a ver qué técnica usamos para diagnosticar en estas situaciones y cuándo se comienza a perder la capacidad de diagnóstico conforme la exhaustividad del diccionario se reduce.

La hipótesis de la que partimos para diseñar las técnicas de diagnóstico es que los SEU próximos entre sí producen patrones de error similares a la salida. Estos pueden ser caracterizados de diferentes formas y usados para estimar la localización real del SEU que queremos localizar.

Combinando la información que obtenemos al aplicar distintas métricas sobre la información disponible, hemos conseguido unos resultados bastante buenos sobre los diseños en los que se ha probado la técnica. Incluso para aquellos circuitos en los que no se consigue acertar el biestable y ciclo exactos, la técnica, tras la primera iteración, acota la localización del SEU en un rango relativamente reducido de ciclos y a unos registros concretos. A partir de esta primera acotación podemos obtener un nuevo diccionario de fallos enfocado en las zonas del circuito señaladas por el algoritmo de diagnóstico y repetir con él el proceso, mejorando el resultado. El diagnóstico puede darse por finalizado cuando encontremos un candidato que produzca exactamente el mismo patrón de salida que el SEU bajo diagnóstico.

Con este proceso iterativo, si el diccionario de partida es lo suficientemente completo para realizar correctamente la primera estimación, llegará un momento en el que podamos obtener un diccionario completo de la zona acotada. Si llegados a este punto aún no ha terminado el diagnóstico y las iteraciones han seguido el camino correcto, el siguiente diccionario contendrá, al menos, una entrada cuyo patrón de salida coincida con el patrón que produce el SEU bajo diagnóstico.

Esta técnica puede ser muy útil en el proceso de diseño de circuitos resistentes a radiación, ya que, por ejemplo, ante cualquier vulnerabilidad encontrada tras irradiar el circuito en el acelerador de partículas, evita repetir el proceso de diseño completo. Aplicando la técnica se puede saber en qué biestable se ha producido el SEU y reforzar la zona en caso de que fuera necesario.

Abstract

The Single Event Upset (SEU) diagnosis is an open problem that has hardly been investigated previously. In this work we seek to design a new diagnostic technique that allows locating a SEU from the information that is available.

It is common to have only incomplete fault dictionaries, since, on large circuits, the time required to obtain a complete fault dictionary makes it unfeasible. We are going to see what technique we use to diagnose in these situations and when the diagnostic capacity begins to lose, according to the exhaustiveness of the dictionary.

The hypothesis from which we start to design diagnostic techniques is that the SEUs close to each other produce similar patterns at the output. These can be characterized in different ways and used to estimate the actual location of the SEU that we want to locate.

Combining the information we obtain by applying different metrics on the available information, we have achieved quite good results on the designs in which the technique has been tested. Even for those circuits in which it is not possible to hit the exact flip-flop and cycle, the technique, after the first iteration, limits the location of the SEU in a relatively reduced range of cycles and to specific registers. From this first dimension we can obtain a new fault dictionary focused on the areas of the circuit indicated by the diagnostic algorithm and repeat the process with it, improving the result. The diagnosis can be terminated when we find a candidate that produces the exact same output pattern as the SEU under diagnosis.

With this iterative process, if the starting dictionary is complete enough to make the first estimate correctly, there will come a time when we can obtain a complete dictionary of the bounded area. If at this point the diagnosis has not yet finished and the iterations have followed the correct path, the following dictionary will contain at least one entry whose output pattern matches the pattern that the diagnostic SEU produces.

This technique can be very useful in the process of designing radiation resistant circuits, since, for example, before any vulnerability found after irradiating the circuit in the particle accelerator, it avoids repeating the entire design process. By applying the technique, it is possible to know in which bistable the SEU has been produced and to reinforce the area if necessary.

... -translation by google-

Abbreviated index

<i>Resumen</i>	III
<i>Abstract</i>	V
<i>Abbreviated index</i>	VII
1 Introducción	1
2 Estado del arte	3
2.1 Detección de fallos (<i>Fault Detection</i>)	3
2.2 Diagnóstico de fallos o localización de fallos	3
3 Conclusiones y trabajos futuros	5
3.1 Conclusiones	5
3.2 Trabajos futuros	6
<i>List of Figures</i>	7
<i>List of Tables</i>	9
<i>List of Codes</i>	11
<i>Bibliography</i>	13
<i>Index</i>	17
<i>Glossary</i>	17

Contents

<i>Resumen</i>	III
<i>Abstract</i>	V
<i>Abbreviated index</i>	VII
1 Introducción	1
2 Estado del arte	3
2.1 Detección de fallos (<i>Fault Detection</i>)	3
2.2 Diagnóstico de fallos o localización de fallos	3
3 Conclusiones y trabajos futuros	5
3.1 Conclusiones	5
3.2 Trabajos futuros	6
3.2.1 Otras técnicas de tratamiento de imágenes	6
3.2.2 Distancia en flip-flops. Mejora de la distancia temporal	6
<i>List of Figures</i>	7
<i>List of Tables</i>	9
<i>List of Codes</i>	11
<i>Bibliography</i>	13
<i>Index</i>	17
<i>Glossary</i>	17

1 Introducción

La primera vez que se observaron los efectos de la radiación en satélites en órbita fue a mediados de la década de 1970. Desde entonces, los investigadores han estudiado sus efectos sobre diferentes circuitos y tecnologías. La radiación puede ser un problema para los circuitos destinados a trabajar en su presencia. Si esta es ionizante, puede dar lugar a un *Single Event Effect (SEE)*, o un efecto de evento singular, provocando un error en el circuito. Los daños que provoca la radiación se clasifican en dos grandes grupos: *errores físicos ('hard errors')* y *errores lógicos ('soft errors')* [1]. Las *conmutaciones por evento singular o Single Event Upset (SEU)* son errores lógicos inducidos por radiación en el circuito que consisten en el cambio de valor de un biestable del mismo. No son daños permanentes, pero sí que pueden afectar al correcto funcionamiento del sistema.

Con la miniaturización de los circuitos, la dosis de radiación necesaria para provocar un SEU es cada vez menor, con la consiguiente aparición de sus efectos cada vez a menor altitud [2]. Esto acerca el problema de la radiación a aplicaciones más comunes como puede ser la aviación o las telecomunicaciones. A veces no importa, o es asumible, que un bit del circuito conmute a causa de radiación. Blindar un móvil frente a radiación o reforzar sus circuitos con técnicas como la *Redundancia Modular Triple o Triple Modular Redundancy (TMR)* [3] puede ser innecesario dado que el mayor riesgo al que nos exponemos es mínimo, pero esto no siempre es así. Cuando se trata de satélites, aviones, o incluso bases militares armadas, existen sistemas críticos cuyas misiones pueden ser el control orbital, la estabilización del vuelo o el lanzamiento de misiles, donde no son asumibles los errores que pueda provocar un SEU.

Diseñar circuitos resistentes a radiación puede ser un proceso costoso, complicado y lento. Además, aplicar técnicas de refuerzo contra radiación a circuitos completos puede ser una pérdida de recursos [4]. Uno de los pasos del diseño suele ser emplear una plataforma de inyección de fallos para estudiar qué zonas del circuito son críticas y cuáles no necesitan ser reforzadas [4]. Si todo ha ido bien, uno de los últimos pasos suele ser irradiar el circuito de forma real para verificar el diseño. Si se detectan irregularidades a la salida a causa de un SEU ocurrido durante la prueba, sería necesario rediseñar el circuito para reforzar aquellas zonas donde se hayan producido las conmutaciones. Este proceso se vería enormemente beneficiado de una técnica que permita localizar los SEU, es decir, calcular el ciclo de reloj y biestable en el que ha tenido lugar. Determinar la localización espacial y temporal de un SEU se denomina *SEU diagnosis* [5].

El problema al que nos enfrentamos al tratar de localizar un SEU a partir de la información de salida de un circuito crece exponencialmente con el tamaño del circuito. Las escasas técnicas de diagnóstico existentes hasta el momento requieren de diccionarios de fallos completos o exhaustivos, requisito que no siempre es posible cumplir.

En la presente investigación hemos desarrollado una nueva técnica de diagnóstico de SEU basada en diccionarios de fallos incompletos o no exhaustivos. La hipótesis de la que partimos es que:

Hipótesis 1.0.1 *"Dos SEU próximos entre sí provocarán patrones de error similares a la salida".*

La mayor parte de la investigación se ha centrado en obtener métricas para examinar los patrones de salida desde distintas aproximaciones, ya que el parecido o no de dos salidas depende mucho de cómo las observemos. En el desarrollo del presente documento analizaremos y compararemos las métricas desarrolladas. Veremos si alguna de ellas es mejor que otras, cómo podemos combinarlas en un único algoritmo que realice el diagnóstico, si este mejora o empeora al prescindir de alguna de las métricas fusionadas, y hasta qué punto podemos elevar la calidad del diagnóstico empleando esta técnica.

2 Estado del arte

2.1 Detección de fallos (*Fault Detection*)

Dado que no es posible realizar un diagnóstico de SEU sin detectarlo primero, numerosos estudios se centran en desarrollar técnicas que permitan detectarlos a tiempo para suprimir sus efectos. Por ejemplo, en 2014, un equipo chino presentó una técnica de detección de SEU basada en la *Máquina de Boltzman Restringida o Restricted Boltzmann Machine (RBM)*, bloque fundamental en muchos algoritmos de *Deep Learning* [6]. En [7] abordan el problema de *fault detection* por el modelo dinámico del sistema. Comparan las lecturas tomadas por los sensores con los valores teóricos que se obtienen del modelo dinámico del robot SCARA. De esta forma detectan anomalías debidas a radiación. En un estudio más reciente, enfocado a sistemas embebidos, emplean programas de detección por software. Multitud de hilos se ejecutan simultáneamente y se encargan de examinar el circuito con el objetivo de detectar alguna irregularidad causada por radiación [8].

2.2 Diagnóstico de fallos o localización de fallos

Hasta ahora, el diagnóstico de fallos ha sido poco estudiado, siendo los fallos de fabricación a los que más esfuerzos de investigación se les ha dedicado [9, 10, 11, 12, 13, 14, 15]. Estos no son el tipo de fallos que nos interesa diagnosticar en esta investigación, ya que no son causados por radiación, si no que se producen, como su nombre indica, en el momento de fabricación del circuito (*stuck-at-0*, *stuck-at-1*).

Las técnicas existentes para localización de fallos provocados por radiación se basan principalmente en el uso de diccionarios de fallos, aunque también se emplean vectores de prueba, listas de fallos, tabla de verdad de nodos ("*node truth table*") y tabla de conexiones de pines (*pin connection table*) [16, 17, 18].

A excepción de contados estudios, la mayoría de los revisados modelan al circuito bajo prueba o *Circuit Under Test (CUT)* como una caja negra, es decir, el diseño del circuito no se conoce y solo las salidas pueden ser monitorizadas. Normalmente, el número de biestables del circuito es mucho mayor que el número de salidas, por lo que es necesario observar el circuito el suficiente tiempo como para detectar patrones que puedan ser asociados a la localización de un determinado SEU [5]. Estas huellas son registradas y almacenadas en un diccionario durante una fase previa al diagnóstico.

El diccionario de fallos se genera mediante inyección de fallos, en alguna plataforma que lo permita [19, 20, 21], y contiene información de la localización de los SEU inyectados y el patrón de salidas que produce. Si el diccionario recoge todas las posibilidades, se habla de diccionario completo o exhaustivo, tomando el nombre de la campaña de inyección de fallos necesaria para generarlo (*Campaña Exhaustiva*). En el caso contrario, es un diccionario incompleto o no exhaustivo,

es decir, no todas las posibles combinaciones de (biestable, ciclo) han sido inyectadas y almacenadas en el diccionario.

Durante la fase de diagnóstico, para localizar un SEU detectado, se compara el patrón de error que genera en las salidas del circuito con los patrones almacenados en el diccionario. Debido al largo tiempo de observación comentado, la información a comparar puede tener un tamaño considerable, y por tanto el tiempo necesario para procesar la comparación es alto. Una solución para reducir esta cantidad de información y por tanto, el tiempo, permitiendo incluso localización de SEU en tiempo real, es comprimirla. Un ejemplo sería el uso de códigos HASH [5].

Dada la gran cantidad de biestables existentes en comparación con el reducido número de salidas, no es difícil imaginar la posibilidad de que dos SEU localizados en biestables y/o ciclos distintos produzcan exactamente el mismo patrón de error a la salida, al menos durante el tiempo y test programados. Cuando esto ocurre, se habla de "*Colisión*". Además, es posible que un SEU no produzca error alguno a la salida durante el test, siendo indistinguible de una situación libre de conmutaciones. Ante estas situaciones, existirá más de una entrada del diccionario que coincida con la búsqueda. Como resultado del diagnóstico se obtienen no una si no una lista de posibles localizaciones para el SEU bajo diagnóstico.

Hasta ahora hemos hablado de diagnóstico empleando diccionarios de fallos completos, pero si el CUT es grande, obtener un diccionario exhaustivo es una operación inviable, ya que la cantidad de combinaciones biestable-ciclo a inyectar para ello se vuelve inabarcable. Si se intenta diagnosticar un SEU empleando un diccionario de fallos incompleto, aparecen nuevos problemas, ya que puede ocurrir que la ubicación correcta no se haya inyectado durante la prueba, y por tanto no se encuentre en el diccionario. Si además existe una colisión que sí se ha inyectado, el diagnóstico concluirá con una localización única aparentemente correcta que puede no se acerque nada a la real.

3 Conclusiones y trabajos futuros

3.1 Conclusiones

El propósito de esta investigación era desarrollar técnicas de diagnóstico de SEU utilizando diccionario de fallos incompletos. Para ello, partíamos de la hipótesis 1.0.1, marcando la línea de investigación en encontrar una métrica adecuada que permita sacar partido de las similitudes en los patrones que generan dos conmutaciones lógicas próximas entre sí.

Hemos realizado experimentos en los que se diagnosticaba aplicando métricas cuyos enfoques eran diferentes, todos ellos devolviendo unos resultados muy buenos. Además, hemos establecido las bases de una técnica de diagnóstico consistente en la realización de numerosas iteraciones de *"Campaña de inyección de fallos para generar un diccionario de fallos"*, *"Diagnóstico empleando el diccionario de fallos obtenido"*, *"Extracción de candidatos que acoten el espacio de inyección"*, con la cuál hemos podido realizar diagnósticos exitosos con diccionarios de fallos muy incompletos en circuitos de grandes dimensiones. Además, observamos como en muchas ocasiones, el diagnóstico se resuelve con la campaña inicial, ya que los candidatos obtenidos no dejan duda.

Se ha propuesto además que siempre se realice una campaña extra tras finalizar el diagnóstico, ya que los resultados apuntan a que iteraciones extras obtendrían, de darse el caso, otras colisiones extras que enriquecerían el diagnóstico.

La técnica de campañas iterativas puede llevarnos hacia un mínimo local donde no se encuentre el SEU bajo diagnóstico, fallando su misión. Hemos propuesto soluciones a este problema tales como la inclusión de inyecciones aleatorias en cada iteración.

Hemos comprobado que la distancia temporal es especialmente útil para localizar temporalmente el SEU bajo diagnóstico. Para circuitos grandes, con diccionarios muy reducidos, la exactitud de la distancia en ciclos se reduce, pero los resultados demuestran que es capaz de acotar bastante la localización temporal del SEU.

La distancia de Levenshtein funciona especialmente bien como métrica de diagnóstico. Hemos comprobado que a distancias de Levenshtein bajas, el cálculo de ciclo realizado con la distancia temporal (ecuación ??) mejora, aunque no es un requisito imprescindible para que este cálculo funcione con exactitud.

Los experimentos encaminados a determinar cuándo se pierde la capacidad de diagnóstico sobre un circuito a base de reducir progresivamente la exhaustividad del diccionario muestran que ésta no se pierde instantáneamente. Primero perdemos la capacidad de localizar el biestable exacto, y posteriormente se van encontrando más dificultades para determinar el registro donde se localiza el SEU, siendo la capacidad de diagnóstico temporal lo último que se pierde. Es en este punto de pérdida total donde deja de tener potencial de diagnóstico completo la técnica iterativa, ya que hasta entonces, existirán campañas que mejoren el resultado.

El hecho de que el primer intento de encontrar una distancia útil para diagnóstico haya resultado muy bien, nos lleva a pensar que podrían existir muchas otras métricas aún por explorar y que funcionen incluso mejor. Podemos concluir la investigación afirmando que estas técnicas tienen capacidad de diagnóstico con diccionarios de fallos incompletos; especialmente la técnica de campañas iterativas, la cual mantiene capacidad de diagnóstico incluso con diccionarios de fallos muy incompletos.

3.2 Trabajos futuros

Las mejoras más inmediatas que se podrían realizar sobre la técnica son optimizaciones computacionales. El código está realizado completamente en Python, por ser un lenguaje de programación con el que se hacen scripts muy rápidamente. Traducir el lenguaje a código C reduciría significativamente el coste computacional y los tiempos de cómputo.

Para optimizar el uso de la memoria, se podría reemplazar la forma de almacenar los daños de cada entrada del diccionario por tablas dispersas.

Como trabajo futuro más inmediato, se podría aplicar la técnica a circuitos de mayor extensión. Además sería interesante aplicar la técnica en circuitos tras realizarles tests de radiación.

Por último, se podrían investigar otras formas de explotar las similitudes existentes entre patrones cercanos.

3.2.1 Otras técnicas de tratamiento de imágenes

El campo de la percepción por ordenador está muy desarrollado actualmente y avanza muy rápido. Sería interesante explorar qué otras opciones existen para diagnosticar basándonos en el tratamiento de imágenes.

Además, el *deep learning* está experimentando grandes avances que podrían ser aplicados para entrenar una red neuronal capaz de localizar un SEU del circuito para la que se le ha entrenado.

3.2.2 Distancia en flip-flops. Mejora de la distancia temporal

Una posible mejora del diagnóstico, basada en la topología del circuito, puede ser la que hemos llamado "*Distancia en FF*".

La distancia en flip-flop (FF) de un biestable a otro sería el número mínimo de ciclos de reloj que tomaría propagar un SEU de uno a otro. Esta distancia puede verse afectada por la dirección:

$$D(A - B) \neq D(B - A)$$

Calculamos que esta distancia puede ser útil para afinar el diagnóstico una vez ya se haya acotado lo suficiente al SEU.

Para calcular la distancia en FF habría que contar cuántos elementos de lógica combinacional separan dos biestables. Esto podría realizarse empleando la suite de Yosys por ejemplo [22].

List of Figures

List of Tables

List of Codes

Bibliography

- [1] H. G. Miranda, “Aportaciones a las técnicas de emulación y protección de sistemas microelectrónicos complejos bajo efectos de la radiación,” Ph.D. dissertation, Universidad de Sevilla, May 2010.
- [2] M. Santarini, “Cosmic radiation comes to asic and soc design,” May 2005. [Online]. Available: <https://www.edn.com/cosmic-radiation-comes-to-asic-and-soc-design/>
- [3] C. Carmichael, “Triple module redundancy design techniques for virtex fpgas,” *Xilinx Application Note XAPP197*, vol. 1, 2001.
- [4] M. G. Valderas, M. P. García, C. López, and L. Entrena, “Extensive seu impact analysis of a pic microprocessor for selective hardening,” in *2009 European Conference on Radiation and Its Effects on Components and Systems*, 2009, pp. 333–336.
- [5] J. M. Mogollón, J. Nápoles, H. Guzmán-Miranda, and M. A. Aguirre, “Real time seu detection and diagnosis for safety or mission-critical ics using hash library-based fault dictionaries,” in *2011 12th European Conference on Radiation and Its Effects on Components and Systems*, 2011, pp. 705–710.
- [6] S. Jian, J. Jiang, K. Lu, and Y. Zhang, “Seu-tolerant restricted boltzmann machine learning on dsp-based fault detection,” in *2014 12th International Conference on Signal Processing (ICSP)*, 2014, pp. 1503–1506.
- [7] W. Tao and W. Xingsong, “Fault diagnosis of a scara robot,” in *2008 15th International Conference on Mechatronics and Machine Vision in Practice*, 2008, pp. 352–356.
- [8] R. Pettit and A. Pettit, “Detecting single event upsets in embedded software,” in *2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC)*, 2018, pp. 142–145.
- [9] B. K. Sikdar, N. Ganguly, and P. P. Chaudhuri, “Fault diagnosis of vlsi circuits with cellular automata based pattern classifier,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 7, pp. 1115–1131, 2005.
- [10] S. S. Yau and Yu-Shan Tang, “An efficient algorithm for generating complete test sets for combinational logic circuits,” *IEEE Transactions on Computers*, vol. C-20, no. 11, pp. 1245–1251, 1971.
- [11] S. S. Yau and M. Orsic, “Fault diagnosis and repair of cutpoint cellular arrays,” *IEEE Transactions on Computers*, vol. C-19, no. 3, pp. 259–262, 1970.

- [12] V. Amar and N. Condulmari, "Diagnosis of large combinational networks," *IEEE Transactions on Electronic Computers*, vol. EC-16, no. 5, pp. 675–680, 1967.
- [13] D. R. Schertz and G. Metze, "A new representation for faults in combinational digital circuits," *IEEE Transactions on Computers*, vol. C-21, no. 8, pp. 858–866, 1972.
- [14] J. P. Roth, W. G. Bouricius, and P. R. Schneider, "Programmed algorithms to compute tests to detect and distinguish between failures in logic circuits," *IEEE Transactions on Electronic Computers*, vol. EC-16, no. 5, pp. 567–580, 1967.
- [15] A. D. Friedman, "Fault detection in redundant circuits," *IEEE Transactions on Electronic Computers*, vol. EC-16, no. 1, pp. 99–100, 1967.
- [16] Su Wei, Fan Tongshun, and Du Mingfang, "Research for digital circuit fault testing and diagnosis techniques," in *2009 International Conference on Test and Measurement*, vol. 1, 2009, pp. 330–333.
- [17] S. Wei, Z. Shide, and X. Lijun, "Research on digital circuit fault location procedure based on lasar," in *2008 ISECS International Colloquium on Computing, Communication, Control, and Management*, vol. 2, 2008, pp. 322–326.
- [18] N. Naber, T. Getz, Y. Kim, and J. Petrosky, "Real-time fault detection and diagnostics using fpga-based architectures," in *2010 International Conference on Field Programmable Logic and Applications*, 2010, pp. 346–351.
- [19] R. Zhang, L. Xiao, J. Li, X. Cao, C. Qi, and M. Wang, "A fast fault injection platform of multiple seus for sram-based fpgas," in *2017 Prognostics and System Health Management Conference (PHM-Harbin)*, 2017, pp. 1–5.
- [20] A. da Silva and S. Sanchez, "Leon3 vip: A virtual platform with fault injection capabilities," in *2010 13th Euromicro Conference on Digital System Design: Architectures, Methods and Tools*, 2010, pp. 813–816.
- [21] J. M. Mogollon, H. Guzmán-Miranda, J. Nápoles, J. Barrientos, and M. A. Aguirre, "Ftunshades2: A novel platform for early evaluation of robustness against see," in *2011 12th European Conference on Radiation and Its Effects on Components and Systems*, 2011, pp. 169–174.
- [22] C. Wolf, J. Glaser, and J. Kepler, "Yosys-a free verilog synthesis suite," in *Proceedings of the 21st Austrian Workshop on Microelectronics (Austrochip)*, 2013.
- [23] Zhou Jing, Liu Zengrong, Chen Lei, Wang Shuo, Wen Zhiping, Chen Xun, and Qi Chang, "An accurate fault location method based on configuration bitstream analysis," in *NORCHIP 2012*, 2012, pp. 1–5.
- [24] Wikipedia, "Distancia de levenshtein — wikipedia, la enciclopedia libre," 2020, [Internet; descargado 15-junio-2020]. [Online]. Available: https://es.wikipedia.org/w/index.php?title=Distancia_de_Levenshtein&oldid=125248609
- [25] M. Muñoz-Quijada, S. Sanchez-Barea, D. Vela-Calderon, and H. Guzman-Miranda, "Fine-grain circuit hardening through vhdl datatype substitution," *Electronics*, vol. 8, no. 1, p. 24, 2019.
- [26] "Vhdl implementation of fft algorithm(s)," Available online: <https://github.com/thasti/fft>, accessed on 17 June 2020.

-
- [27] "Vhdl standard fifo," Available online: <http://www.deathbylogic.com/2013/07/vhdl-standard-fifo/>, accessed on 17 June 2020.
 - [28] "Fpga4student. a low pass fir filter for ecg denoising in vhdl," Available online: <https://www.fpga4student.com/2017/01/a-low-pass-fir-filter-in-vhdl.html>, accessed on 17 June 2020.
 - [29] "I²s interface designed for the pcm3168 audio interface from texas instruments," Available online: <https://github.com/wklimann/PCM3168>, accessed on 17 June 2020.
 - [30] "Simple uart controller for fpga written in vhdl," Available online: <https://github.com/jakubcabal/uart-for-fpga>, acceded on 17 June 2020.
 - [31] Wikipedia, "Momentos de imagen — wikipedia, la enciclopedia libre," 2020, [Internet; descargado 18-junio-2020]. [Online]. Available: https://es.wikipedia.org/w/index.php?title=Momentos_de_imagen&oldid=124767713

Glossary

CUT Circuit Under Test. 3, 4

FF flip-flop. 6

RBM Restricted Boltzmann Machine. 3

SEE Single Event Effect. 1

SEU Single Event Upset. III, 1, 3–6

TMR Triple Modular Redundancy. 1