

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica



**Diseño de un entorno mínimo de pruebas de validación de circuitos integrados en
NI TestStand**

**Informe de Proyecto de Graduación para optar por el título de Ingeniero en
Electrónica con el grado académico de Licenciatura**

César Andrés Ortega Valverde

Cartago, Noviembre 2018

INSTITUTO TECNOLÓGICO DE COSTA RICA

ESCUELA DE INGENIERÍA ELECTRÓNICA

PROYECTO DE GRADUACIÓN

ACTA DE APROBACIÓN

**Defensa de Proyecto de Graduación
Requisito para optar por el título de Ingeniero en Electrónica
Grado Académico de Licenciatura
Instituto Tecnológico de Costa Rica**

El Tribunal Evaluador aprueba la defensa del proyecto de graduación denominado: “Diseño de un entorno mínimo de pruebas de validación de circuitos integrados en NI TestStand”, realizado por el señor César Andrés Ortega Valverde y, hace constar que cumple con las normas establecidas por la Escuela de Ingeniería Electrónica del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal Evaluador



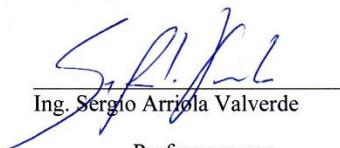
Ing. Luis Carlos Rosales Alpizar

Profesor lector



Ing. Anibal-Ruiz Barquero

Profesor lector



Ing. Sergio Arriola Valverde

Profesor asesor

Cartago, 26 de noviembre, 2018

Declaratoria de autenticidad

Declaro que el presente proyecto de graduación ha sido realizado, en su totalidad, por mi persona y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos que he utilizado material bibliográfico, he procedido a indicar las fuentes mediante citas. En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.



César Andrés Ortega Valverde

1-1528-0940

Noviembre de 2018

Resumen

Los procesos de diseño de circuitos han evolucionado con el pasar de los años, uno de los cambios más drásticos ha sido la reducción del tamaño de los circuitos, pero esta reducción ha traído consigo una gran cantidad de problemas nuevos que han provocado que las etapas de pruebas de circuitos deban mejorar a un ritmo aún más acelerado. Entre las etapas de prueba, la que más se ha visto afectada es la validación post-silicio, donde se comprueba que un chip ya fabricado cumple con todos los requisitos para los que fue diseñado. Debido a ellos los entornos de validación son cada vez más complejos y es necesario que sean más flexibles para mejorar la producción de circuitos.

Este proyecto busca sentar las bases para crear futuros entornos de validación post-silicio, a partir de una metodología de diseño basada en un proceso de ingeniería, que permita no solo crear entornos robustos, sino que también sean hechos a la medida de los circuitos que construya el Laboratorio de Diseño de Circuitos Integrados de la Escuela de Ingeniería Electrónica.

Palabras clave: DFT, DUT, LabVIEW, sistemas empotrados, Test Stand.

Abstract

Circuits size reduction has been one of the most drastic changes over the years in circuit design and a handful of new problems along with; situation that is demanding circuit tests to be improved at an even faster pace. Circuit tests process consist on some stages and the one that has been most affected is the post-silicon validation, where a manufactured chip is verified, defining whether meets all the requirements for which it was designed for. Hence, validation environments are increasingly complex and need to be more flexible to improve circuit production.

This work describes the process of establishing some preliminary foundations to create future post-silicon validation environments, based on a design methodology with an engineering process, which allows not only to create robust environments, but also to be custom made for circuits that are built at Integrated Circuits Design Laboratory at Electronics Engineering School.

Keywords: DFT, DUT, LabVIEW, embedded system, Test Stand.

Dedicatoria

A mi madre Ana y a mi papá José, por todos los sacrificios que hicieron para permitirme cumplir con mi carrera, por haber estado ahí siempre que los necesité y cuyas palabras de aliento me dieron la confianza para seguir adelante.

A mi hermana Jennifer y a mi hermano David, por ser ejemplo de profesionales y por todos los consejos que me brindaron en busca de obtener este título.

A mis amigos de la universidad, Juliana, Adriana, Gustavo, Javier y muchos otro más que me acompañaron a lo largo de este camino y que siempre estuvieron ahí para apoyarme y trabajar juntos en equipo.

A mis primeros jefes en el ámbito laboral don Héctor y don Giancarlo, porque me guiaron y me aconsejaron en tomar la decisión de ingresar a esta universidad.

Agradecimientos

Al profesor Ing. Sergio Arriola Valverde. por toda la ayuda brindada en este proceso de tesis, guiarme en obtener los mejores resultados posibles y aprender muchas cosas que serán de mucha utilidad en la siguiente etapa de mi vida.

Al profesor Ing. Roberto Molina Robles. por permitirme ser parte de este proyecto, la confianza brindada a lo largo de todo ese proyecto y los consejos que ayudaron a completar este trabajo.

A la Escuela de Ingeniería en Electrónica, por facilitarme todas las herramientas necesarias para completar este proyecto.

Al señor Bryan Aguirre, mi jefe en LAIMI, por permitirme ser parte de esa familia y haberme dando grandes lecciones de vida y muchos buenos consejos que siempre tendré conmigo.

A todos los profesores y personas, que a lo largo de mi vida han ayudado a construir el profesional que soy hoy.

Índice General

| | | |
|----------|---|-----------|
| 1 | Introducción | 1 |
| 1.1 | Objetivos y Estructura del documento | 1 |
| 2 | Validación Post-Silicio con Sistemas Empotrados | 3 |
| 2.1 | Validación Post-Silicio..... | 3 |
| 2.1.1 | Procesos de Validación Post-silicio | 4 |
| 2.2 | Design for Testability (DFT)..... | 5 |
| 2.2.1 | Técnicas Ad-hoc | 5 |
| 2.2.2 | Técnicas Estructurales | 6 |
| 2.3 | Device Under Test (DUT)..... | 8 |
| 2.3.1 | Señales Eléctricas..... | 8 |
| 2.4 | Sistemas Empotrados para Validación Post-Silicio | 9 |
| 2.4.1 | Hardware..... | 9 |
| 2.4.2 | Software | 10 |
| 3 | Metodología de Diseño de un Entorno Mínimo de Validación Post-Silicio | 14 |
| 3.1 | Análisis de requerimientos | 14 |
| 3.2 | Diseño de solución | 15 |
| 3.2.1 | Establecimiento de Requerimientos de Circuito..... | 16 |
| 3.2.2 | Medición y Recopilación de Datos | 20 |
| 3.2.3 | Comparación y Evaluación de Datos | 22 |
| 3.2.4 | Reporte de Resultados..... | 22 |
| 3.3 | Implementación del Sistema Validación Post-Silicio | 24 |
| 4 | Validación y Análisis de Resultados | 26 |
| 4.1 | Hardware y software del entorno de validación..... | 26 |
| 4.2 | Prueba de concepto #1: Compuerta AND 7408..... | 27 |

| | | |
|----------|---|-----------|
| 4.3 | Prueba de Concepto #2: Memoria SRAM 2114..... | 41 |
| 4.4 | Propuesta de entorno de validación para procesador RISC-V | 56 |
| 5 | Conclusiones | 61 |
| 5.1 | Recomendaciones..... | 62 |
| 6 | Bibliografía | 64 |
| 7 | Anexos | 66 |
| A. | Mapeo de pines de los VI's | 66 |
| B. | Diagramas de bloques de los VI's de pruebas..... | 68 |

Índice de Figuras

| | |
|---|----|
| Figura 2.1. Crecimiento del consumo de tiempo del proceso de validación [3] | 3 |
| Figura 2.2. Técnicas DFT clasificadas según los tipos ad-hoc y estructurales [5]..... | 7 |
| Figura 2.3. Representación de una señal analógica [9] | 8 |
| Figura 2.4. Representación de una señal digital lógica [9]..... | 9 |
| Figura 2.5. Chasis NI PXI-1078 con varios módulos de medición | 10 |
| Figura 3.1. Diagrama de bloques de la metodología de diseño propuesta | 15 |
| Figura 3.2. Comparación de metodología con actividades de validación [1]..... | 16 |
| Figura 3.3. Diagrama de bloques del primer bloque de la metodología..... | 17 |
| Figura 3.4. Ejemplo de la tabla de alimentación de una hoja de datos [15] | 18 |
| Figura 3.5. Ejemplo de la tabla de características eléctricas de una hoja de datos [15] | 18 |
| Figura 3.6. Ejemplo de tabla de tiempos de una hoja de datos [16] | 19 |
| Figura 3.7. Ejemplo de diagrama de tiempos de una hoja de datos [16]..... | 19 |
| Figura 3.8. Diagrama de bloques del segundo módulo de la metodología..... | 20 |
| Figura 3.9. Ejemplo de encabezado de reporte de validación | 23 |
| Figura 3.10. Ejemplo de tabla de reporte de resultados de validación | 23 |
| Figura 3.11. Comparación de la metodología de diseño con el entorno final | 25 |
| Figura 4.1. Distribución interna de pines de la compuerta AND [15]..... | 27 |
| Figura 4.2. Diagrama de flujo de las pruebas eléctricas de la compuerta AND..... | 28 |
| Figura 4.3. Indicador numérico de la prueba AND VCC | 29 |
| Figura 4.4. Indicadores numéricos de la prueba AND OUTPUT VOLT TEST | 29 |
| Figura 4.5. Diagrama de flujo de la prueba AND TRUTHTABLE TEST | 30 |
| Figura 4.6. Front Panel de la prueba AND TRUTHTABLE TEST para un resultado correcto | 31 |
| Figura 4.7. Front Panel de la prueba AND TRUTHTABLE TEST para un resultado incorrecto | 31 |
| Figura 4.8. Diagrama de flujo de la secuencia de pruebas de la compuerta AND | 32 |
| Figura 4.9. Secuencia de pruebas en TestStand para la compuerta AND | 33 |
| Figura 4.10. Variables locales de TestStand para la secuencia de la compuerta AND | 33 |
| Figura 4.11. Reporte de resultados para el caso de fallo en la fuente DC | 34 |
| Figura 4.12. Regla de comparación para la prueba AND VCC TEST | 34 |

| | |
|---|----|
| Figura 4.13. Reporte de resultados para el caso de fallo de la prueba AND VCC TEST | 35 |
| Figura 4.14. Reglas de comparación para la prueba AND OUTPUT VOLTAGE TEST | 35 |
| Figura 4.15. Reglas de comparación para la prueba AND TRUTHTABLE TEST | 36 |
| Figura 4.16. Resumen del reporte de resultados de la compuerta marca Motorola para los 3 grupos de iteraciones | 37 |
| Figura 4.17. Resumen del reporte de resultados de la compuerta marca TI para los 3 grupos de iteraciones | 37 |
| Figura 4.18. Resumen del reporte de resultados de la compuerta marca Renesas para los 3 grupos de iteraciones | 38 |
| Figura 4.19. Reporte de resultados de la prueba AND VCC TEST | 39 |
| Figura 4.20. Reporte de resultados de la prueba AND OUTPUT VOLTAGE TEST | 40 |
| Figura 4.21. Reporte de resultados de la prueba AND TRUTHTABLE TEST | 40 |
| Figura 4.22. Diagrama de pines de la memoria 2114 [16] | 41 |
| Figura 4.23. Diagramas de flujo de las pruebas eléctricas de la memoria SRAM | 42 |
| Figura 4.24. Front panel de la prueba SRAM VCC TEST | 43 |
| Figura 4.25. Front Panel de la prueba SRAM OUTPUT VOLT TEST | 43 |
| Figura 4.26. Diagrama funcional de la prueba SRAM MEMORY TEST | 44 |
| Figura 4.27. Front panel de la prueba SRAM MEMORY TEST para el caso correcto | 45 |
| Figura 4.28. Front panel de la prueba SRAM MEMORY TEST para el caso incorrecto | 46 |
| Figura 4.29. Diagrama de flujo de la secuencia de pruebas de la memoria SRAM | 47 |
| Figura 4.30. Secuencia de pruebas en TestStand para la memoria SRAM | 48 |
| Figura 4.31. Variables locales de TestStand para la secuencia de la memoria SRAM | 48 |
| Figura 4.32. Reporte de resultados para el caso de fallo en la fuente DC | 49 |
| Figura 4.33. Regla de comparación de la prueba SRAM VCC TEST | 49 |
| Figura 4.34. Reporte de resultados para el caso de fallo de la prueba SEAM VCC TEST .. | 50 |
| Figura 4.35. Reglas de comparación para la prueba SRAM OUTPUT VOLT TEST | 50 |
| Figura 4.36. Reglas de comparación de la prueba SRAM MEMORY TEST | 51 |
| Figura 4.37. Resumen del reporte de resultados de la memoria marca National Semiconductor para los 3 grupos de iteraciones..... | 52 |
| Figura 4.38. Resumen del reporte de resultados de la memoria marca NEC Electronics para los 3 grupos de iteraciones..... | 52 |

| | |
|--|----|
| Figura 4.39. Resumen del reporte de resultados de la memoria marca Synertek para los 3 grupos de iteraciones | 53 |
| Figura 4.40. Resumen del reporte de resultados para el caso de medición en función de la velocidad de reloj..... | 54 |
| Figura 4.41. Gráfico de duración de la secuencia SRAM en función de la cantidad de iteraciones..... | 56 |
| Figura 4.42. Diagrama del bloque de comunicación para pruebas del TOP-LEVEL | 57 |
| Figura 4.43. Ejemplo de set de instrucciones aritméticas para el CORE | 58 |
| Figura 4.44. Diagrama de bloques del entorno de validación para el TOP-LEVEL..... | 58 |
| Figura 4.45. Diagrama de flujo del módulo de escritura para el procesador..... | 59 |
| Figura 4.46. Diagrama de flujo para el módulo de lectura del procesador..... | 60 |
| Figura B.1. Diagrama de bloques del VI para encender la fuente DC del PXI..... | 68 |
| Figura B.2. Diagrama de bloques del VI para apagar la fuente DC del PXI..... | 69 |
| Figura B.3. Diagrama de bloques del VI AND VCC TEST..... | 69 |
| Figura B.4. Diagrama de bloques del VI AND OUTPUT VOLT TEST | 69 |
| Figura B.5. Diagrama de bloques del VI AND TRUTH TABLE TEST | 70 |
| Figura B.6. Diagrama de bloques del VI SRAM VCC TEST | 70 |
| Figura B.7. Diagrama de bloques del VI SRAM OUTPUT VOLT TEST, parte 1 | 71 |
| Figura B.8. Diagrama de bloques del VI SRAM OUTPUT VOLT TEST, parte 2..... | 71 |
| Figura B.9. Diagrama de bloques del VI SRAM MEMORY TEST, parte 1 | 72 |
| Figura B.10. Diagrama de bloques del VI SRAM MEMORY TEST, parte 2 | 73 |
| Figura B.11. Diagrama de bloques del VI SRAM MEMORY TEST, parte 2 | 74 |

Índice de Tablas

| | |
|--|----|
| Tabla 2.1. Tipos de Datos [10] | 12 |
| Tabla 3.1. Tabla de requerimientos mínimos del entorno de validación mínimo | 14 |
| Tabla 4.1. Especificaciones de hardware y software de la plataforma NI PXI | 26 |
| Tabla 4.2. Especificaciones de las herramientas de software del entorno..... | 26 |
| Tabla 4.3. Parámetros eléctricos de la compuerta AND [15] | 27 |
| Tabla 4.4. Tabla de Verdad de la Compuerta AND [15]..... | 28 |
| Tabla 4.5. Resultados de la secuencia AND en función de la cantidad de iteraciones | 38 |
| Tabla 4.6. Tabla de datos eléctricos de la memoria 2114 [16] | 41 |
| Tabla 4.7. Tabla de verdad de las funciones de la memoria 2114 [16] | 42 |
| Tabla 4.8. Resultados de secuencia SRAM según cantidad de iteraciones | 53 |
| Tabla 4.9. Resultados de secuencia SRAM según frecuencia de reloj..... | 55 |
| Tabla 4.10. Predicción de duración de secuencia según iteraciones | 55 |
| Tabla A.1. Distribución de pines de la prueba AND VCC..... | 66 |
| Tabla A.2. Distribución de pines de la prueba AND OUTPUT VOLTAGE TEST..... | 66 |
| Tabla A.3. Distribución de pines de la prueba AND TRUTHTABLE TEST | 66 |
| Tabla A.4. Distribución de pines de la prueba SRAM VCC | 67 |
| Tabla A.5. Distribución de pines de la prueba SRAM OUTPOT VOLT TEST | 67 |
| Tabla A.6. Distribución de pines de la prueba SRAM MEMORY TEST..... | 67 |

1 Introducción

Actualmente la Escuela de Ingeniería en Electrónica se encuentra en desarrollo de un proyecto que busca realizar un SoC (System on Chip), que incluye un CPU basado en la arquitectura RISC-V HV. En este proyecto participan el Laboratorio de Diseño de Circuitos Integrados (DCILab) de la Escuela de Ingeniería Electrónica, el grupo microDIE de la Universidad Católica del Uruguay, el grupo OnchipUIS de la Universidad Industrial de Santander en Colombia, y el Centro Médico Erasmus en Rotterdam, Países Bajos.

En todo proceso de desarrollo de un chip se tiene la etapa de validación post-silicio que es el último paso antes de poder dar como finalizado el diseño de un circuito, En la validación post-silicio es posible determinar fallos que se hayan desarrollado a lo largo de la fabricación del circuito y en algunos casos permite encontrar fallos no detectados en la etapa de verificación pre-silicio.

En relación con lo anterior, actualmente el DCILab no cuenta con ningún entorno de validación post-silicio o alguna plataforma estructurada que les permita realizar las pruebas necesarias para encontrar fallos en circuitos fabricados, por el Laboratorio.

La solución propuesta para este proyecto busca que en el DCILab se utilice el equipo y las herramientas que ya se tienen a disposición, con el fin de crear su propio entorno de validación. Para generar este entorno se usarán las herramientas de software NI LabVIEW para generar los módulos de recopilación de datos de las diferentes características de los chips, NI Test Stand para automatizar el proceso de ejecución de las pruebas, la validación de los datos obtenidos y el reporte de los resultados de la validación del circuito y la plataforma de hardware NI PXI para realizar las mediciones de las características de los circuitos.

1.1 Objetivos y Estructura del documento

El objetivo general de este trabajo es implementar un entorno de validación mínimo, con las herramientas antes descritas, que pueda ser utilizado para validar circuitos de prueba. Para lograr la implementación del entorno mínimo, será necesario desarrollar una metodología de diseño para entornos de validación. Una vez definida la metodología, se diseñarán los

circuitos a nivel funcional mediante la utilización de VI's programados en LabVIEW, los cuales permitirán además recopilar datos de características eléctricas y funcionales de los circuitos. Finalmente se llevará a cabo una validación del entorno mínimo mediante pruebas de concepto que permitan contrastar el funcionamiento y robustez del entorno, contemplando además un análisis de plataforma que permita clarificar los alcances y limitaciones que poseen las herramientas de desarrollo utilizadas, durante la ejecución de cada prueba.

El presente informe está compuesto por cinco capítulos, distribuidos de la siguiente manera: el Capítulo 2 expone los conceptos básicos y necesarios para entender el proceso de diseño del entorno mínimo, además de conceptos de sistemas empotrados y herramientas de software para validación. En el Capítulo 3 se describe el diseño de una metodología para conformación de un entorno de validación y luego su implementación en herramientas de software/hardware. El Capítulo 4 muestra los resultados obtenidos a partir de pruebas de concepto realizadas bajo diferentes condiciones y criterios de pruebas. Finalmente, en el Capítulo 5 se muestran las conclusiones y recomendaciones del trabajo llevado a cabo, generado un hilo conductor para trabajos a futuros.

2 Validación Post-Silicio con Sistemas Empotrados

En este capítulo se expondrán conceptos básicos y necesarios para apoyar el entendimiento de los temas que componen este trabajo. No obstante, si desea profundizar en relación a temas específicos, se recomienda consultar las referencias [1-16].

2.1 Validación Post-Silicio

Validación post-silicio se entiende como el conjunto de pruebas, mediciones y actividades dedicadas a comprobar que un circuito integrado (CI) fabricado, cumple con todas las condiciones y características eléctricas según se establecieron al momento de dar inicio su fabricación. El objetivo de esta etapa es asegurarse de que el CI se desempeña correctamente bajo condiciones de operación reales, mientras ejecuta diferentes tareas para la cuales fue diseñado. [1]

El proceso de validación comienza con el primer CI fabricado y termina cuando se da el visto bueno para su producción en masa. La validación en la mayoría de los procesos de diseño de CI's, es capaz de ocupar hasta el 70% del tiempo y además puede representar un consumo de recursos según las pruebas realizadas. No obstante, al mismo ritmo que se ha ido reduciendo el tamaño de fabricación en los chips, ha ido aumentando el tiempo que consume un proceso de validación en al menos un 35% de todo el ciclo de diseño de un chip. [2]

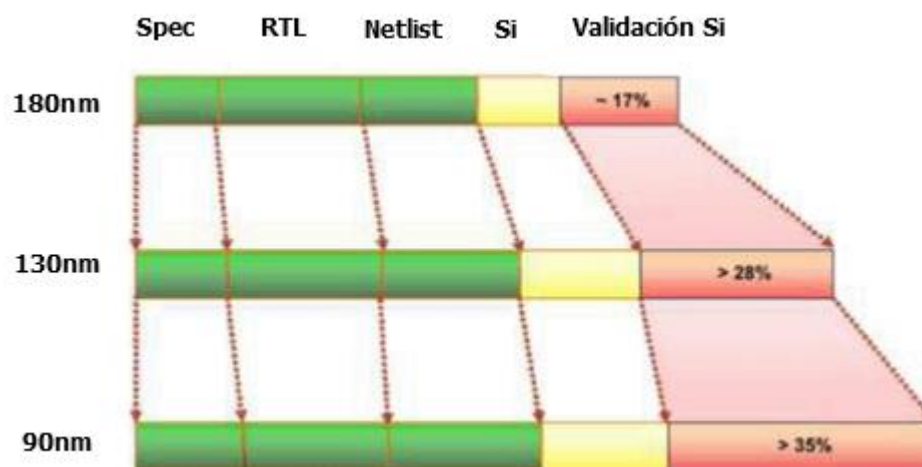


Figura 2.1. Crecimiento del consumo de tiempo del proceso de validación [3]

A pesar de la gran cantidad de recursos que consume esta etapa, el hecho de realizarse en un chip ya fabricado permite no solo realizar las pruebas de funcionamiento lógico que se verían en un ambiente de verificación pre-silicio, sino que también permite comprobar características no-funcionales como el consumo de potencia, estrés de trabajo, tolerancia de temperatura y margen de ruido eléctrico. En contraposición, encontrar un error en esta etapa significa que se debe realizar una gran inversión de recursos para corregir dicho fallo, además de que no se podrían hacer modificaciones al chip en el momento para mejorar los controles, característica que si posee la verificación pre-silicio al tener la posibilidad de modificar el código. [3]

2.1.1 Procesos de Validación Post-silicio

Existen múltiples actividades a realizar en la etapa de validación post-silicio a continuación, se resumen las actividades más comunes de esta etapa: [1]

Power-on-debug: Debe ser la primera actividad por realizar en el proceso de validación debido a que un correcto arranque y alimentación de este no solo permitirá desarrollar todas las pruebas necesarias, sino que también le brindará al chip un estado de operación correspondiente a lo que encontrará al salir al mercado. Sin una correcta alimentación y arranque, no se podrá tener acceso la arquitectura funcional del chip además de que, en la mayoría de los casos, no se podrá tener ninguna visibilidad y controlabilidad del diseño interno. Completar esta actividad procede típicamente con la construcción de una placa anti-fallos donde se puede colocar el chip y este sea alimentado de manera correcta.

Basic Hardware Logic Validation: Busca comprobar que el diseño de hardware funciona correctamente, y además garantiza de poner a prueba bloques específicos del diseño. Se compone de pruebas dirigidas a tareas específicas del diseño como también a eventos aleatorios. En resumen, en este proceso se evalúan todas las características funcionales del circuito.

Hardware/Software Compatibility Validation: Comprueba la compatibilidad del diseño con diferentes versiones de los sistemas, aplicaciones y periféricos donde será implementado el diseño. Debido a la gran cantidad de posibles combinaciones de configuraciones, este proceso puede consumir una gran porción de tiempo al tener que comprobar todas y cada una de las combinaciones.

Electrical Validation: Se enfoca en comprobar las características eléctricas del chip y todo lo relacionado con él, con el objetivo de presentar resultados favorables e incluso en el peor caso posible de condiciones de operación. Aquí se prueban las entradas-salidas (I/O), consumo de potencia, velocidad de reloj y otras pruebas analógicas y digitales.

Speed-Path Validation: En todos los circuitos integrados, el rendimiento en términos de velocidad del chip estará definido por la velocidad de propagación de datos de la ruta más lenta en el chip. Como los circuitos están compuestos de millones de transistores que pueden llegar a tener una diferente velocidad de respuesta entre ellos, todas las rutas del circuito pueden tener velocidades de transferencias de datos diferentes. El objetivo de esta actividad es encontrar esa ruta más lenta de manera que se pueda optimizar el diseño lo más posible.

2.2 Design for Testability (DFT)

Se entiende por DFT como “cualquier estrategia que permita incrementar la controlabilidad y observabilidad de los puntos internos de un circuito” [4]. Controlabilidad se define como la capacidad de aplicar estímulos de prueba en un punto determinado del circuito, mientras que la observabilidad se comprende como la capacidad de monitorizar externamente las respuestas a tales estímulos. En relación con lo anterior, se puede determinar qué el testability de un circuito en términos de la controlabilidad y observabilidad de sus estructuras internas. Las técnicas de DFT se pueden dividir en dos categorías, ad-hoc y estructurales. Las técnicas estructurales se pueden subdividir en dos grupos según la metodología utilizada, técnicas de scan-path y técnicas built-in-selftest (BIST). [5]

2.2.1 Técnicas Ad-hoc

Las técnicas ad-hoc consisten en métodos heurísticos que se enfocan en probar las estructuras internas del circuito, por medio del manejo de sus entradas y salidas, para su control y observación desde el exterior. Estas técnicas son diseñadas en función de circuitos específicos por lo que es muy difícil que sean replicables para otros circuitos. Por ser un método dedicado, al final se consideran a estas técnicas como reglas prácticas para el DFT, en función de mejorar la controlabilidad y la observabilidad de un circuito. [5]

2.2.2 Técnicas Estructurales

Al contrario de las técnicas ad-hoc, las técnicas estructurales buscan seguir una metodología de diseño que puedan ser aplicables de manera general a cualquier circuito. No obstante, la técnica ad-hoc tiende a tener un mejor rendimiento, pero con el defecto de aumentar el costo económico del circuito ya que se debe agregar circuitería extra que permita realizar las pruebas de manera interna, aumentando el área del chip y sus pines. Estas técnicas se pueden dividir entre técnicas de scan-path y técnicas built-in-selftest (BIST).

Técnicas Scan-path

Este grupo de técnicas buscan controlar y modificar la operación de los latches del sistema, por medio de registros de desplazamiento que se encargan de controlar y observar los nodos internos del circuito. Este método propone normalmente agregar multiplexores a la entrada de los elementos de memoria del sistema, manejando así los latches al colocar el circuito en un estado de “test” interno. Una de sus ventajas es que se enfoca principalmente en comprobar las partes combinatoriales del circuito al aplicar vectores de pruebas y de simulación cuando el circuito se encuentra en modo de “test”, separando de manera lógica la parte combinatorial y secuencial del circuito, además de que particiona el circuito en función de la cantidad de scan-paths utilizados. Aunque no permite encontrar específicamente la ubicación del fallo, los valores extraídos de los registros de desplazamiento dan pistas sobre la sección de lógica combinatorial donde puede estar el fallo. Una de sus mayores desventajas es que se necesitan una gran cantidad de ciclos de testeo por lo que aumentan el consumo y los retardos en el circuito. [5]

Técnicas Built-in-self-test (BIST)

Las técnicas BIST agregan una lógica extra al circuito que se encarga de generar internamente los vectores de pruebas y al mismo tiempo, hacer las comparaciones respectivas con los resultados obtenidos. Este método de validación beneficia al reducir algunos costes económicos por la generación externa de los vectores y simplificar el uso de equipo externo, además permite probar los circuitos en condiciones normales de operación sin afectar los tiempos de retardo o el consumo. Pero, debido a que es necesario crear un espacio propio para la lógica de comprobación, aumentan significativamente el área de los circuitos más que cualquier otra técnica y debido a que las respuestas de comprobación son almacenadas en

memoria, se dificulta mucho el poder comprobar elementos específicos del circuito ya que esto aumentaría mucho la cantidad de memoria del circuito y por ende el área de este. Por ello esta técnica en lugar de comprobar los espacios del vector uno a uno, ingresa toda la respuesta a una máquina de estados finitos donde uno de sus estados es una bandera de “correcto” mientras que el resto son estados de “incorrecto”. De esta manera al ir comprobando esta máquina la respuesta, si se presenta algún fallo, la máquina termina en un estado de “incorrecto”, en caso contrario la bandera sería la del estado “correcto”. [5]

Debido a la creciente complejidad de los circuitos, no basta con aplicar solo una de las técnicas antes descritas, por lo que el diseñador debe conocer todas las técnicas disponibles y eso lleva a que hoy en día los límites que las diferencian ya no sean tan claros. Esto ha llevado a que se creen nuevas técnicas que resultan ser híbridos de las 3 principales antes descritas. La figura 2.2 muestra cómo están distribuidas la mayoría de las técnicas DFT y como se relacionan entre ellas:

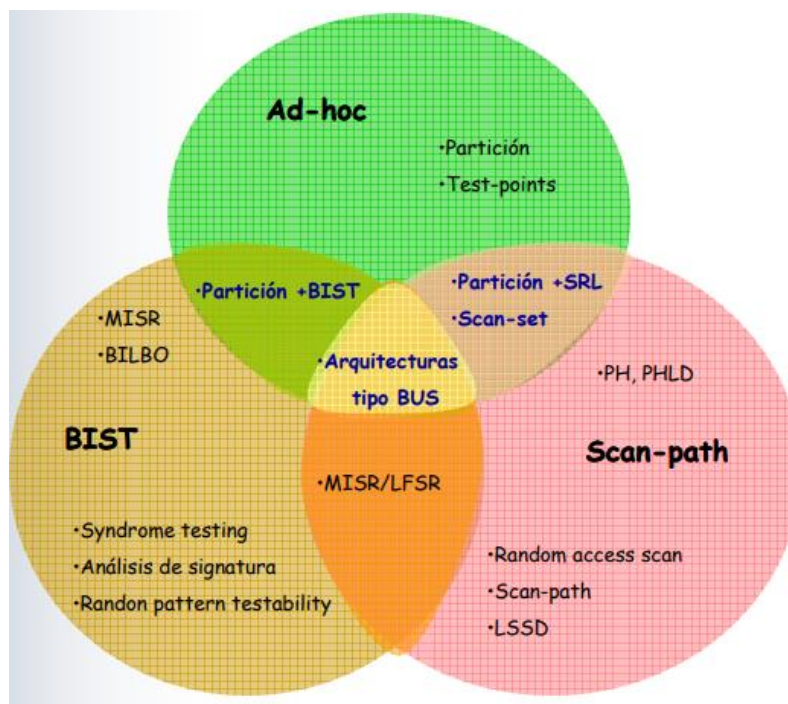


Figura 2.2. Técnicas DFT clasificadas según los tipos ad-hoc y estructurales [5]

2.3 Device Under Test (DUT)

Un Dispositivo Bajo Prueba (DUT, por sus siglas en inglés) se refiere a un producto de manufacturación que está siendo sometido a diferentes pruebas para determinar su estado o su rendimiento. También se le conoce como Unidad Bajo Prueba (UUT, por sus siglas en inglés) o Circuito Bajo Prueba (CUT, por sus siglas en inglés) dependiendo del producto que se esté probando. [6] Todos estos dispositivos por lo general utilizan periféricos para conectarse con otros dispositivos, la comunicación con estos dispositivos externos es por medio de señales eléctricas, ya sean analógicas o digitales.

2.3.1 Señales Eléctricas

Una señal se define matemáticamente como una función de una o más variables, usada para el transporte de información. Las señales pueden ser catalogadas según la variable de dicha función, en el caso de las señales eléctricas esta variable sería el tiempo por lo que se dividirían en señales de tiempo continuo y señales de tiempo discreto. [7]

En electrónica, se asocian las señales de tiempo continuo a las señales analógicas, las cuales son aquellas cuyos valores varían de forma continua a lo largo del tiempo. Todos aquellos fenómenos físicos que pueden ser percibidos como la luz, la temperatura, el sonido, la presión, son considerados señales analógicas. Como varían de forma continua, en un espacio de tiempo pueden tomar una infinita cantidad de valores. Su forma característica es la de una onda senoidal. [8] La figura 2.3 muestra un ejemplo de una señal analógica.

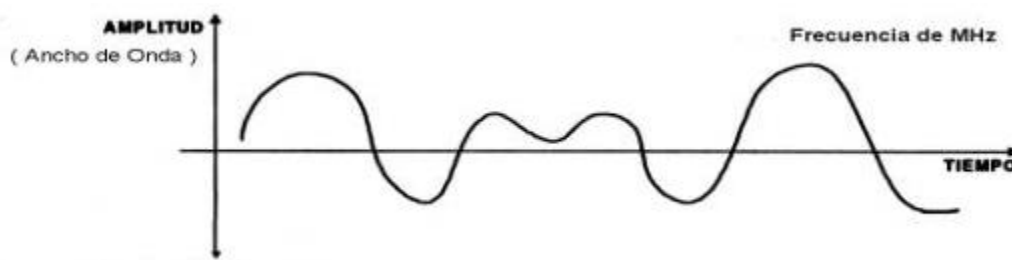


Figura 2.3. Representación de una señal analógica [9]

Las señales de tiempo discreto son aquellas que son definidas solo en intervalos de tiempo específicos. En electrónica se les define con las señales digitales, las cuales son discretas tanto en tiempo como en amplitud por lo que solo pueden tomar una cantidad finita de valores a diferencia de las señales analógicas. Las señales digitales pueden ser obtenidas al procesar señales analógicas por medio de sistemas llamados Convertidores Analógicos-Digitales

(ADC). Su forma característica es la de una onda cuadrada y se representan en el dominio del tiempo. Las señales lógicas utilizadas en electrónica son señales digitales cuyos límites de amplitud están definidos en forma binaria, es decir, solo pueden tomar los valores de 1 o 0. [7] La figura 2.4 muestra un ejemplo de señal digital.

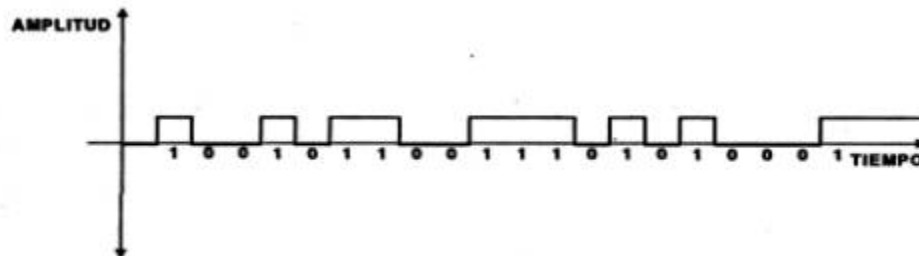


Figura 2.4. Representación de una señal digital lógica [9]

2.4 Sistemas Empotrados para Validación Post-Silicio

Un sistema empotrado o embebido hace referencia a todos aquellos sistemas que combinan hardware y software para realizar tareas específicas, dentro de un sistema o dispositivo más grande con funciones más generales. Las funciones de los sistemas empotrados son comúnmente las de control dentro de sistemas más grandes, aunque en algunos casos también se encargan de realizar mediciones y recopilaciones de datos. A pesar de que surgen de combinar hardware con software y muchos poseen sistemas operativos, se diferencian de las computadoras personales ya que su parte de software está limitada a realizar solo tareas específicas con el fin de consumir menos recursos y hacer lo más compacto posible el sistema. [11]

Los sistemas empotrados dedicados a la validación post-silicio utilizan de igual manera equipo de hardware para hacer las mediciones, y software especializado para recopilar esos datos y comprobarlos. En este proyecto se combinan el hardware NI PXI y los softwares NI LabVIEW y NI TestStand:

2.4.1 Hardware

NI PXI

El NI PXI es una plataforma basada en PC especializada en mediciones y automatización de sistemas que combina la tecnología de bus PCI, la tecnología de empaquetado modular CompactPCI de Eurocard y buses de sincronización especializados con el fin de realizar

tareas de medición para pruebas, adquisición de datos y aplicaciones de manufactura. Se compone de 3 partes básicas: el chasis, el controlador de sistema y los módulos periféricos. [12]

El chasis proporciona la estructura robusta de empaquetado modular que, gracias a una backplane, permite manejar varios buses de comunicación sincronizados entre ellos. El controlador de sistema representa el cerebro del PXI y permite manejar la plataforma como una computadora volviéndola así independiente de sistemas externos, aunque también existen controladores con características remotas para dirigir la plataforma desde otras computadoras, servidores o incluso otros PXI. Los módulos periféricos permiten tener una gran gama de equipos de medición y automatización, existen cerca de 200 módulos creados por la propia compañía y más de 1000 creados por compañías externas. [12]

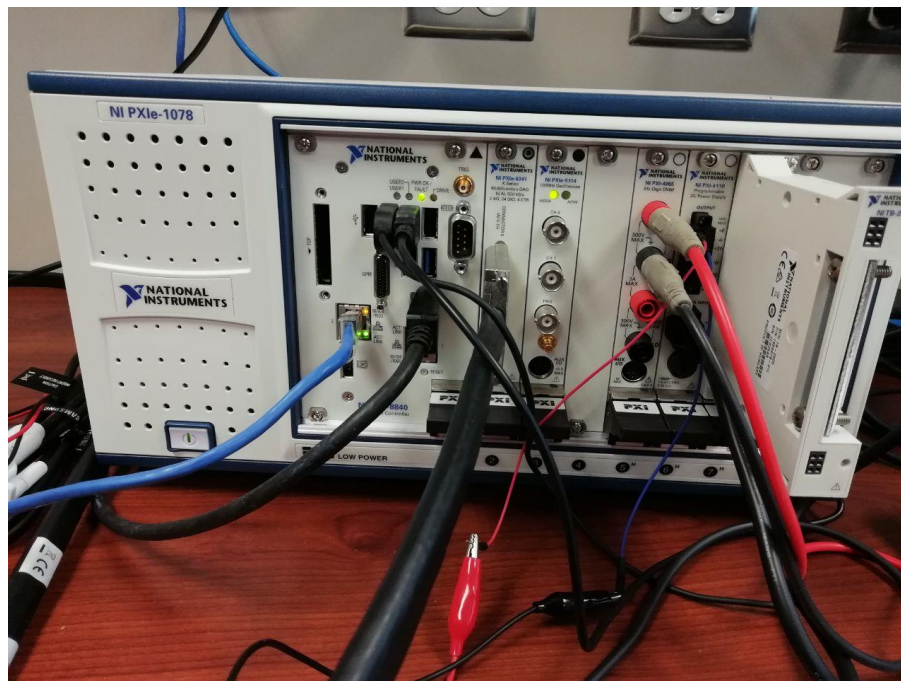


Figura 2.5. Chasis NI PXI-1078 con varios módulos de medición

2.4.2 Software

NI LabVIEW

NI LabVIEW es un software de programación gráfica que ayuda a visualizar cada aspecto de una aplicación, incluyendo configuración de hardware, datos de medidas y depuración. Esta visualización hace que sea más fácil integrar hardware de medidas de cualquier

proveedor, representar una lógica compleja en el diagrama, desarrollar algoritmos de análisis de datos y diseñar interfaces de usuario personalizadas. [13]

Las pruebas específicas estarán compuestas por Instrumentos Virtuales (que en adelante serán mencionados como VI's por sus siglas en inglés, Virtual Instruments), estos son los programas de LabVIEW y su apariencia y operación simulan a los instrumentos físicos, como osciloscopios y multímetros. Por medio de la extensa variedad de herramientas para adquirir, analizar, visualizar y almacenar datos que posee LabVIEW, se podrán desarrollar una gran variedad de pruebas que serán fáciles de manipular y que podrán ser editadas a conveniencia. [13]

NI TestStand

NI TestStand es un software de administración de pruebas que está diseñado para desarrollar más rápido sistemas de pruebas automatizadas y validación. Además, puede desarrollar secuencias de pruebas que integran módulos de código escritos en cualquier lenguaje de programación de pruebas. Las secuencias también especifican flujo de ejecución, registro de base de datos y conectividad a otros sistemas empresariales. Además, puede desplegar sistemas de pruebas para producción con interfaces de operador fáciles de usar. [14]

El Editor de Secuencias de TestStand permite crear secuencias que automatizan la ejecución de módulos de pruebas de diferentes fuentes de programación. El editor posee 4 tipos de evaluaciones que son las que se asocian a los módulos de pruebas, estas evaluaciones son las que determinan como se leen los datos y como se evalúan a partir de reglas de comparación. Al final de la ejecución de la secuencia, TestStand genera automáticamente un reporte en formato XML donde se resumen todos los resultados de las evaluaciones de la secuencia. [14]

Tipos de Datos

Para la validación correcta de un circuito, es importante conocer bien el tipo de dato de la salida que será analizada, por ello en la siguiente tabla se exponen los tipos de datos más comunes y que son utilizados en los procesos de validación:

Tabla 2.1. Tipos de Datos [10]

| Tipo de dato | Subtipo | Descripción | Ejemplo |
|--------------|------------|---|-----------------------------|
| Numérico | Entero | Abreviados como tipo “int”, se refieren a aquellos datos numéricos que no poseen decimales. Se utilizan en los casos en el que el dato no se puede fraccionar. Estos valores pueden o no tener signo, por lo que pueden dividirse también en categoría “signed” o “unsigned” respectivamente. | 5 -67 +43 |
| | Reales | Abreviado normalmente como “float”, debido al uso del punto flotante para representar valores racionales muy grandes. Este tipo de dato si se puede fraccionar por lo que si poseen una parte decimal. | 3,5 +7,4 -5,2 |
| Texto | Caracteres | Abreviado como “char”, representar todos los caracteres conocidos como lo son los alfanuméricos y especiales. En algunos lenguajes de programación se utilizan las comillas simples (‘’) para diferenciarlos de otros tipos de datos. | ‘3’ ‘C’ ‘%’ |
| | Cadenas | Conocido como “string”, se definen como cadenas de caracteres ya sean una palabra, | “Video” “Diseño Digital” |

| Tipo de dato | Subtipo | Descripción | Ejemplo |
|--------------|-----------|---|-----------------------------------|
| | | <p>una frase o una oración. En algunos lenguajes de programación se utilizan las comillas dobles (“”) para diferenciarlos de otros tipos de datos.</p> | <p>“Análisis de Sistemas”</p> |
| Lógicos | Booleanos | <p>Conocidos como tipo “bool”, son datos lógicos y como tal solo pueden tener dos valores: verdadero (true) o falso (false). También pueden identificarse con los números 1 y 0 para el caso verdadero y el caso falso respectivamente. Se utilizan generalmente como banderas, para identificar si se realizó o no un proceso.</p> | <p>True False 1 0</p> |

3 Metodología de Diseño de un Entorno Mínimo de Validación Post-Silicio

En este capítulo se expondrá la metodología de diseño propuesta, utilizando un flujo de ingeniería en el cual se pretende una abstracción de alto nivel de la metodología. Posteriormente, se implementará la metodología en un ambiente LabVIEW-TestStand con la plataforma de hardware NI PXI.

3.1 Análisis de requerimientos

La tabla 3.1 muestra los requerimientos mínimos del entorno de validación y en los que se basó el diseño de esta metodología.

Tabla 3.1. Tabla de requerimientos mínimos del entorno de validación mínimo

| Requerimientos Mínimos |
|--|
| Validar DUT's con un máximo de 24 entradas/salidas digitales |
| Validar DUT's con un máximo de 16 salidas analógicas |
| Validar DUT's con un máximo de 2 entradas analógicas |
| Realizar pruebas de validación de las características eléctricas y funcionales del circuito integrado. |
| Realizar todas las pruebas en una misma ejecución |
| El entorno permite agregar las pruebas correspondientes al circuito integrado que se desea validar. |
| Todas las pruebas seleccionadas deberán ejecutarse de manera automática |
| Conformación de un reporte técnico al final de la prueba. |

3.2 Diseño de solución

La figura 3.1 muestra el diagrama de bloques que describe la metodología de diseño propuesta:

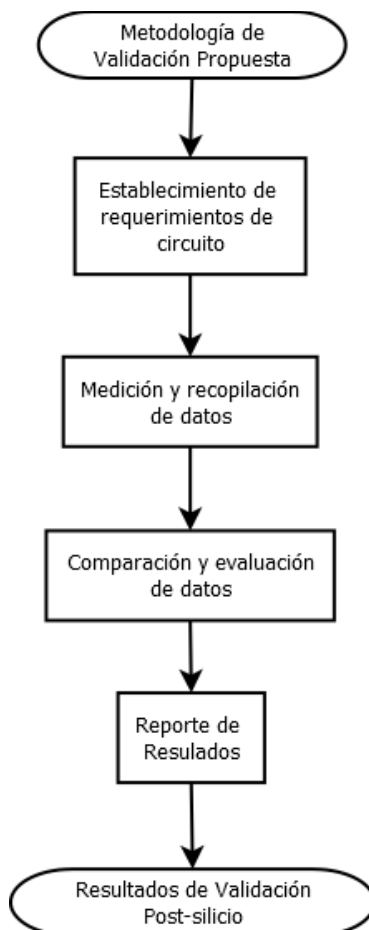


Figura 3.1. Diagrama de bloques de la metodología de diseño propuesta

El desarrollo de la metodología descrita en la figura 3.1 está fundamentada y adaptada a trabajos relacionados tales como el documento “Post-Silicon Validation in the SoC Era: A Tutorial Introduction” [1]. Basado en lo antes descrito, la Figura 3.2 muestra cuales actividades de las expuestas por el documento, influenciaron en el diseño de esta metodología. Todas estas actividades son explicadas en el capítulo 2 de este trabajo. Las actividades “Hardware/Software Compatibility Validation” y “Speed-path Validation” no tienen participación en el diseño debido a que no son parte de los requerimientos solicitados y pueden existir algunas limitaciones para su ejecución.

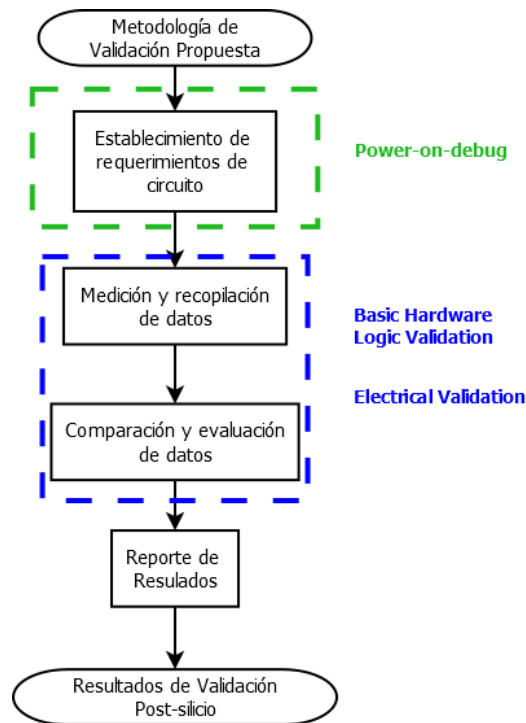


Figura 3.2. Comparación de metodología con actividades de validación [1]

Este entorno está diseñado en función de realizar pruebas catalogadas como “Caja Gris”, debido a que estas facilitan la trazabilidad de errores en los circuitos bajo pruebas. Esto en función a que, previo a la validación, para generar los casos de prueba es necesario conocer el funcionamiento o resultados de los circuitos según los estímulos proporcionados al ser este un entorno de validación post-silicio hecho a medida.

3.2.1 Establecimiento de Requerimientos de Circuito

La figura 3.3 muestra el flujo de procesos para establecer los requerimientos del circuito que será evaluado. Cada proceso explica el análisis que se debe realizar para definir cómo debe funcionar el circuito en un estado de operación normal y como se deben evaluar sus parámetros eléctricos y funcionales.

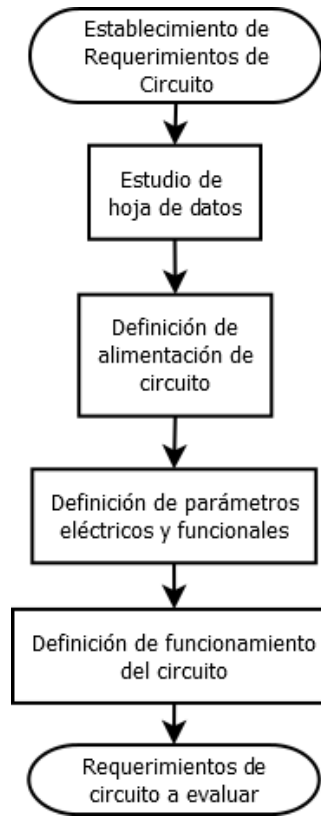


Figura 3.3. Diagrama de bloques del primer bloque de la metodología

Inicialmente se debe hacer un estudio de la hoja de datos del circuito, ya que es donde se definen la alimentación y los valores para la operación del circuito (VCC, Vout, Vin, tablas de verdad, etc.). Las hojas de datos contienen tablas donde se exponen los límites de los valores eléctricos ya sea de alimentación o de operación, también se muestran tablas de verdad o diagramas de tiempo para la operación de las partes combinacionales y secuenciales respectivamente.

Como indica la actividad “Power-on-debug” [1], es recomendable primero definir la alimentación del circuito para asegurar el arranque correcto del chip a probar y que los datos obtenidos sean equivalentes a los del circuito funcionando en un ambiente de operación normal como el que tendrá una vez en el mercado. La figura 3.4 muestra un ejemplo de cómo se presenta esta información en una hoja de datos, normalmente en formato de tabla indicando los valores mínimos, máximos y típicos de los parámetros, además de sus unidades. Existen casos donde esta información no se muestra en una tabla, sino que los fabricantes la ubican al inicio de la hoja de datos junto con otros datos necesarios para la operación en formato de viñetas.

Recommended Operating Conditions

| Item | Symbol | Min | Typ | Max | Unit |
|-----------------------|-----------|------|------|------|-------------|
| Supply voltage | V_{CC} | 4.75 | 5.00 | 5.25 | V |
| Output current | I_{OH} | — | — | -400 | μA |
| | I_{OL} | — | — | 8 | mA |
| Operating temperature | T_{opr} | -20 | 25 | 75 | $^{\circ}C$ |

Figura 3.4. Ejemplo de la tabla de alimentación de una hoja de datos [15]

Para realizar la validación de un circuito, se debe definir los parámetros eléctricos y funcionales que se van a evaluar. La información de las características eléctricas se define normalmente en las hojas de datos en formato de tablas donde se indican el nombre de la variable o una abreviación, los valores mínimos, máximos y típicos, las unidades y en algunos casos se muestra información de las condiciones en que se realizaron estas mediciones.

La figura 3.5 muestra un ejemplo de cómo se presenta esta información en una hoja de datos. Para las características eléctricas se pueden analizar: tensión y corriente de entrada y salida, velocidad de reloj, temperatura, consumo de potencia e incluso en algunos casos como las compuertas, la corriente que se entrega dependiendo de si la salida es un uno o un cero lógico.

Electrical Characteristics

($T_a = -20$ to $+75$ $^{\circ}C$)

| Item | Symbol | min. | typ.* | max. | Unit | Condition |
|------------------------------|-----------|------|-------|------|---------|--|
| Input voltage | V_{IH} | 2.0 | — | — | V | |
| | V_{IL} | — | — | 0.8 | V | |
| Output voltage | V_{OH} | 2.7 | — | — | V | $V_{CC} = 4.75$ V, $V_{IH} = 2$ V, $I_{OH} = -400$ μA |
| | V_{OL} | — | — | 0.5 | V | $I_{OL} = 8$ mA $V_{CC} = 4.75$ V, $V_{IL} = 0.8$ V |
| — | | — | 0.4 | | | |
| Input current | I_{IH} | — | — | 20 | μA | $V_{CC} = 5.25$ V, $V_I = 2.7$ V |
| | I_{IL} | — | — | -0.4 | mA | $V_{CC} = 5.25$ V, $V_I = 0.4$ V |
| | I_I | — | — | 0.1 | mA | $V_{CC} = 5.25$ V, $V_I = 7$ V |
| Short-circuit output current | I_{OS} | -20 | — | -100 | mA | $V_{CC} = 5.25$ V |
| Supply current | I_{CCH} | — | 2.4 | 4.8 | mA | $V_{CC} = 5.25$ V |
| | I_{CCL} | — | 4.4 | 8.8 | mA | $V_{CC} = 5.25$ V |
| Input clamp voltage | V_{IK} | — | — | -1.5 | V | $V_{CC} = 4.75$ V, $I_{IN} = -18$ mA |

Note: * $V_{CC} = 5$ V, $T_a = 25^{\circ}C$

Figura 3.5. Ejemplo de la tabla de características eléctricas de una hoja de datos [15]

Para las características funcionales se debe conocer el comportamiento del circuito, en función de sus valores de entrada y salida. Parte de esta información es proporcionada generalmente por la hoja de datos de fabricante, por ejemplo, las tablas de verdad cuando el

circuito posee una parte combinacional o diagramas de tiempos con sus respectivas tablas cuando posee una parte secuencial.

Para los datos de circuitos secuenciales se tienen tablas que presentan los tiempos de ejecución con sus valores máximos y mínimos y las unidades, los diagramas de tiempos muestran el orden en que se ejecutan las acciones según los tiempos indicados en las tablas. Las figuras 3.6 y 3.7 presentan ejemplos de estos datos respectivamente.

| Symbol | Parameter | 2114-1, 2114L-1 | | 2114-2, 2114L-2 | | 2114-3, 2114L-3 | | 2114, 2114L | | Unit |
|------------------|---------------------------------|-----------------|------|-----------------|------|-----------------|------|-------------|------|------|
| | | Min. | Max. | Min. | Max. | Min. | Max. | Min. | Max. | |
| Read Cycle | | | | | | | | | | |
| t _{RC} | Read Cycle Time | 150 | | 200 | | 300 | | 450 | | nsec |
| t _A | Access Time | | 150 | | 200 | | 300 | | 450 | nsec |
| t _{CO} | Chip Select to Output Valid | | 70 | | 70 | | 100 | | 120 | nsec |
| t _{CX} | Chip Select to Output Enabled | 20 | | 20 | | 20 | | 20 | | nsec |
| t _{OTD} | Chip Deselect to Output Off | 0 | 60 | 0 | 60 | 0 | 80 | 0 | 100 | nsec |
| t _{OHA} | Output Hold From Address Change | 50 | | 50 | | 50 | | 50 | | nsec |

Figura 3.6. Ejemplo de tabla de tiempos de una hoja de datos [16]

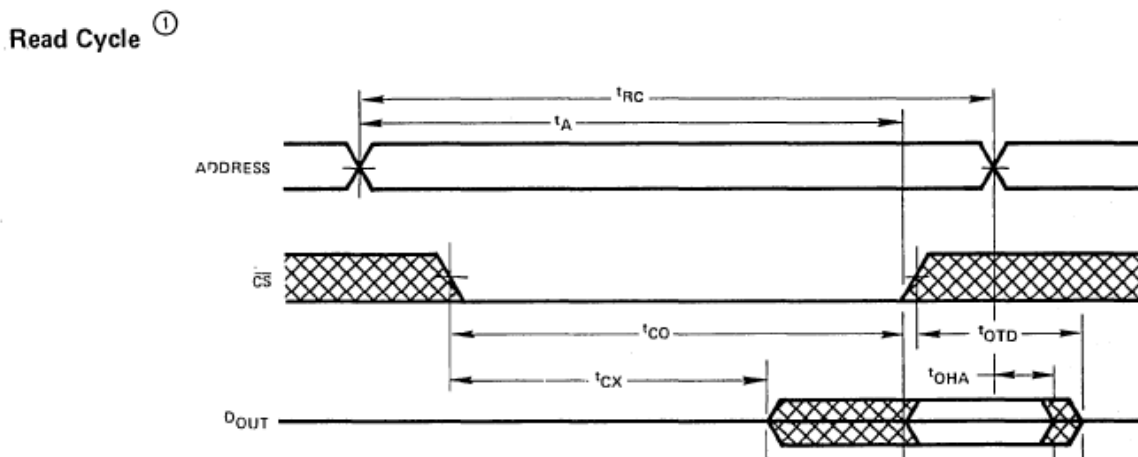


Figura 3.7. Ejemplo de diagrama de tiempos de una hoja de datos [16]

Finalmente, es necesario definir el funcionamiento del circuito o las tareas que este realiza a partir de los datos antes obtenidos, esta información se obtiene también por medio del diseño RTL del circuito, sobre todo en aquellos circuitos que todavía no se han fabricado. Este proceso va en función de ayudar en el diseño del módulo que se encargará de recopilar los datos funcionales del circuito.

Terminado este proceso, se contará con los datos eléctricos y funcionales que permitirán diseñar los módulos de medición y recopilación de datos para la validación del circuito.

3.2.2 Medición y Recopilación de Datos

En esta etapa se explica el proceso para el diseño de los módulos que se encargarán de las mediciones de los parámetros del circuito y su recopilación para su posterior evaluación. Esta etapa está basada en parte en las actividades “Basic Hardware Logic Validation” y “Electrical Validation” [1]. La figura 3.8 muestra el flujo de proceso para la etapa de mediciones y recopilación de datos de la metodología.

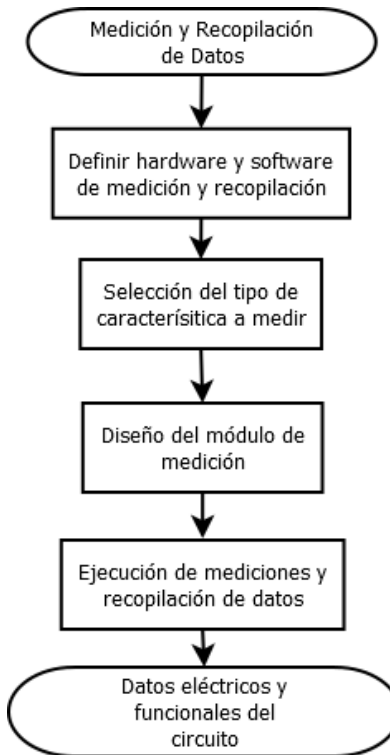


Figura 3.8. Diagrama de bloques del segundo módulo de la metodología

De manera inicial, este proceso requiere de la determinación del hardware y software que será utilizado para llevar a cabo todas las pruebas del circuito, en función de cuál es el parámetro y su tipo.

Los parámetros que se medirán son de tipo eléctrico o funcional, para el caso del primero, la mayoría de sus señales son analógicas por lo que se necesitará equipo indicado para estas mediciones como multímetros, osciloscopios o fuentes regulables que además de alimentar el circuito, pueden generar las señales analógicas de entrada necesarias para obtener las salidas que se deben medir.

Para los parámetros de tipo funcional, en su mayoría se necesitan módulos generadores de señales digitales que permitan ejecutar las tareas de las secciones combinacionales del circuito y para las secciones secuenciales se necesitan módulos como contadores o generadores de frecuencias para controlar los tiempos de reloj.

Definido el equipo a utilizar, se continua con la selección del tipo de parámetro a medir. Este proceso se realiza debido a que las características eléctricas y funcionales no se pueden medir en una misma tarea para evitar conflictos de datos o equipos, por esta razón el documento de ejemplo menciona estas tareas como actividades separadas. [1] En función de lo antes descrito, esta etapa de medición y recopilación de datos se debe repetir por separado para cada parámetro que se debe medir.

Para el diseño de los módulos de medición se debe contar con la información obtenida en la primera etapa de la metodología, para determinar que salidas debe medir el módulo y que entradas debe generar para obtener las salidas correspondientes. Estos módulos se deben categorizar igual como se hizo en los dos procesos anteriores de esta etapa, en función de que el parámetro a medir será eléctrico o funcional.

Los módulos para medición de parámetros eléctricos se basan principalmente en asignar señales analógicas a las entradas del circuito y medir lo que se obtiene a la salida. Existen algunos parámetros eléctricos que solo se pueden medir si a las entradas del circuito se les asignó previamente señales digitales que ejecuten las funciones o tareas del circuito, estos parámetros son en su mayoría las tensiones y corrientes de salida.

Para medir los parámetros funcionales, el módulo debe hacer que el circuito ejecute o realice las tareas para las que fue diseñado. La información del diseño RTL y las tablas de verdad y tiempos del circuito son necesarias para el diseño de estos módulos. Se debe aclarar que, en el caso de los parámetros funcionales, a mayor cantidad de características y tareas realiza el circuito, mayor es el desarrollo de los módulos que recopilan estos datos.

Finalmente, con el módulo terminado, se realizan las mediciones y la recopilación de los datos de las salidas del circuito. Esta etapa solo se puede ejecutar una vez para cada parámetro por lo que, una vez recopilada la información, se repite el proceso para el siguiente parámetro.

Los datos recopilados en este proceso son los que serán evaluados en la siguiente etapa de la metodología.

3.2.3 Comparación y Evaluación de Datos

En esta etapa se deben definir las reglas que determinan si el dato obtenido es correcto o no. Las reglas de comparación son determinadas en función de la información obtenida en la primera etapa de la metodología y son diferentes para cada parámetro según sea eléctrico o funcional, por lo que estas reglas también se categorizan según el tipo de parámetro.

Para los parámetros eléctricos, las reglas son los valores límites de máximo y mínimo indicados en las hojas de datos según corresponda. Para los parámetros funcionales, sus reglas son definidas según las salidas previamente indicadas en las tablas de verdad o en el diseño RTL del circuito. Debe tomarse en cuenta que las salidas pueden tener formatos diferentes como los explicados en la tabla de tipos de datos del capítulo 2.

En función del tipo de dato de la salida, si es un dato numérico, se debe definir el rango de valores numéricos donde la comparación valida que el parámetro es correcto. La mayoría de las salidas de los módulos de medición de parámetros eléctricos son de este tipo. Los módulos de medición de parámetros funcionales tienen en su mayoría salidas de tipo booleanas o de cadena de caracteres por lo que, en el caso del primero, se debe definir cuál de sus estados es el que valida que el dato es correcto. Para el caso que la salida sea una cadena de caracteres, las reglas son más variadas y dependerán del funcionamiento del circuito y de cómo se haya recopilado el dato, la regla más común es que la cadena de salida sea exactamente igual a la cadena de referencia.

3.2.4 Reporte de Resultados.

El reporte final de la validación deber estar compuesto de dos partes, primero el encabezado donde estará toda la información general como la fecha y hora de ejecución de la prueba, el número de serie del circuito bajo prueba, la duración de la prueba, la cantidad de resultados o mediciones, y el resultado final de la validación completa (aprobada o reprobada). La figura 3.9 muestra un ejemplo de este encabezado para el reporte.

| | |
|-------------------------------|-------------------------------|
| Fecha | Sábado, 20 de octubre de 2018 |
| Hora | 15:56:40 |
| Operador de prueba | César Ortega |
| Número de Serie | 00001 |
| Tiempo de Ejecución | 3.57454 segundos |
| Cantidad de Resultados | 15 |
| Resultado Final | Aprobado |

Figura 3.9. Ejemplo de encabezado de reporte de validación

La segunda parte vendría a ser la presentación de los resultados de cada una de las evaluaciones hechas a los datos recopilados. Estos datos pueden ser presentados en un formato de tabla, donde se especifique el dato que se validó, si pasó o no la prueba, el valor de la medición, las unidades de dicha medición y las reglas que se usaron para validar dicha medición. Se deben tomar en cuenta los casos donde se realizan múltiples mediciones de un mismo parámetro, estos resultados deben ser acomodados en un mismo bloque y definir si esta prueba es correcta solo si todas las mediciones cumplieron con las reglas establecidas o no. La figura 3.10 muestra un ejemplo de la tabla de resultados de medición del reporte.

| Prueba | Estado | Valor | Unidad | Límites | | |
|--------------------------|-------------------|--------|--------|---------|--------|---------------------|
| | | | | Mínimo | Máximo | Tipo de Comparación |
| Alimentación de Circuito | Completada | - | - | - | - | - |
| Mediciones: | | | | | | |
| Medición #1 | Aprobada | 4.9449 | V | 4.5 | 5 | GELE (>= <=) |
| Medición #2 | Aprobada | 4.6997 | V | 4.5 | 5 | GELE (>= <=) |
| Medición #3 | Aprobada | 4.6689 | V | 4.5 | 5 | GELE (>= <=) |

Figura 3.10. Ejemplo de tabla de reporte de resultados de validación

El reporte se recomienda que sea almacenado en formato TXT o en formato XML de manera que pueda ser procesador en cualquier sistema o con cualquier programa como MATLAB, Excel, Word, etc. El idioma del reporte es a preferencia del diseñador.

3.3 Implementación del Sistema Validación Post-Silicio

La figura 3.11 muestra en función de las etapas de la metodología, como se implementó con las herramientas de software. La etapa de medición y recopilación de datos fue implementada por medio de LabVIEW, donde se diseñaron VI's dedicados para cada uno de los parámetros de los circuitos. Estos VI's se encargan de controlar los módulos del PXI, realizar las mediciones del circuito y generar las señales necesarias para cada prueba, además cada VI's realiza un pre-procesamiento de los datos en los casos necesarios para que puedan ser evaluados por TestStand.

El diseño de los VI's es en función del parámetro que se medirá, por lo que un VI diseñado para medir las características eléctricas del circuito será muy diferente al VI diseñado para realizar las pruebas de funcionalidad de ese mismo circuito. Aun así, todos los VI's independientemente del tipo de parámetro que se está midiendo deben utilizar la función DAQ Assistant, para lograr generar y recopilar las diferentes señales analógicas y digitales del circuito.

Para cualquier caso donde se plantee utilizar una sección de funcionalidad del circuito o que el parámetro a medir dependa de que el circuito realiza una tarea funcional previa, se necesita dirigir el flujo de ejecución del VI. Las estructuras "Flat Sequence" de LabVIEW permiten dirigir este flujo, por lo que la mayoría de los VI's dedicados a medir las características funcionales del circuito las utilizan.

Los siguientes pasos de la metodología, comparar los datos obtenidos y reportar los resultados, fueron implementados por medio de la herramienta TestStand. En esta herramienta se realizaron las comparaciones y validaciones de los datos recopilados por los VI's por medio de las reglas definidas por la herramienta. La herramienta posee predeterminadamente 4 tipos de "Step Test" donde se evalúan tipos de datos específicos como booleanos, numéricos de una o varias dimensiones y arreglos de caracteres. Dependiendo del dato que envía LabVIEW se selecciona el "Step Test" y este será el que definirá la regla a utilizar. Los límites o los datos de referencia con los que comparará la regla son los definidos en la primera etapa de la metodología, la de determinación de los requerimientos del circuito a evaluar.

Para la generación de los reportes de resultados de validación, TestStand cuenta con su propio generador de reportes interno. El formato de estos reportes es XML, lo que permite que se puedan visualizar en un navegador de internet. Estructuralmente el reporte cuenta con un encabezado con la información general de la prueba y del circuito, luego los resultados son expuestos en el orden de ejecución de la secuencia indicando la medición obtenida y la referencia o regla con que se comparó.

Finalmente, el PXI es la plataforma central donde se ejecuta el entorno, debido a que cuenta con su propio sistema operativo, las herramientas de software se ejecutaron en él y sus módulos fueron los que se encargaron de las mediciones de los parámetros del circuito.

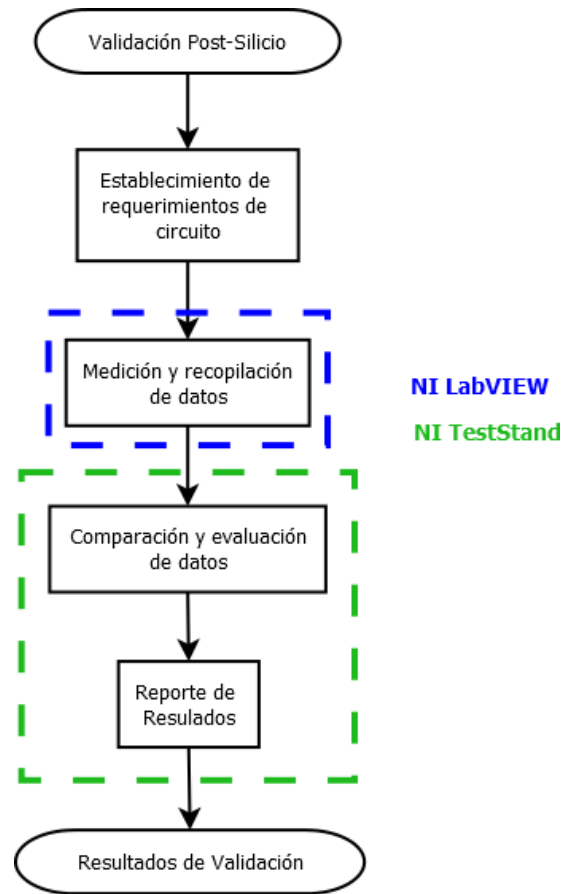


Figura 3.11. Comparación de la metodología de diseño con el entorno final

4 Validación y Análisis de Resultados

En este capítulo se mostrarán los resultados obtenidos en un ambiente de validación diseñado usando la metodología anteriormente expuesta. Se expondrán las 3 pruebas de concepto donde se explicará cómo se diseñó cada una de estas y para cada caso se expondrán los resultados de las mediciones hechas.

4.1 Hardware y software del entorno de validación

Para el desarrollo de entorno de validación y las pruebas de concepto se utilizó la plataforma de hardware NI PXI y las herramientas de software NI LabVIEW y NI TestStand. La tabla 4.1 muestra las especificaciones del hardware de la plataforma PXI.

Tabla 4.1. Especificaciones de hardware y software de la plataforma NI PXI

| Equipo | Modelo/Versión |
|---------------------|--|
| Chasis | NI PXIe-1078 |
| Controlador | NI PXIe-8840 |
| Tarjeta DAQ | NI PXIe-6341 |
| Bloque Conector | SCB-68A |
| Fuente DC Regulable | NI PXI-4110 |
| Sistema Operativo | Windows 7 Professional Edition 64-bits |

La tabla 4.2 muestra las especificaciones de las herramientas de software usadas en el diseño del entorno.

Tabla 4.2. Especificaciones de las herramientas de software del entorno

| Programa | Versión |
|-----------------|----------------|
| NI LabVIEW | v2017 32-bit |
| NI TestStand | v2016 32-bit |

4.2 Prueba de concepto #1: Compuerta AND 7408

Parámetros Eléctricos y Funcionales del Circuito

Para esta prueba se seleccionó la compuerta cuádruple 7408 para analizar el comportamiento del entorno con circuitos de características combinaciones. La información del circuito se obtuvo de su hoja de datos [15], donde se obtuvo la distribución de pines, la tensión de alimentación, su tabla de verdad y las tensiones de salida. En la figura 4.1, se presenta la distribución de pines de la compuerta.

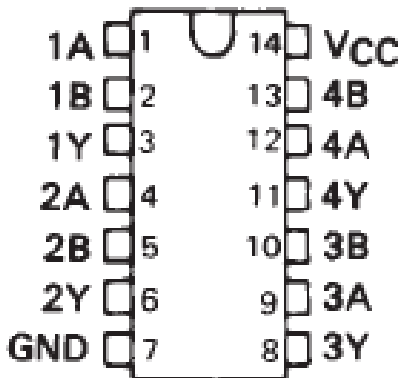


Figura 4.1. Distribución interna de pines de la compuerta AND [15]

La tabla 4.3 muestra los parámetros eléctricos que se evaluaron en el entorno de validación, obtenidos a partir de la hoja de datos del circuito.

Tabla 4.3. Parámetros eléctricos de la compuerta AND [15]

| Parámetro | Valor Mínimo | Valor Típico | Valor Máximo | Unidad |
|-----------|--------------|--------------|--------------|--------|
| VCC | 4.75 | 5 | 5.25 | V |
| VIH | 2 | - | - | V |
| VIL | - | - | 0.8 | V |
| VOH | 2.7 | 3.4 | - | V |
| VOL | - | 0.2 | 0.4 | V |

La tabla 4.4 muestra la tabla de verdad de la compuerta AND, utilizada para validar los parámetros funcionales del circuito.

Tabla 4.4. Tabla de Verdad de la Compuerta AND [15]

| Entrada | | Salida |
|---------|---|--------|
| A | B | Y |
| 1 | 1 | 1 |
| 0 | X | 0 |
| X | 0 | 0 |

Pruebas Eléctricas

Para la validación eléctrica de la compuerta AND, se diseñaron 2 pruebas que midieron y recopilaban los datos de la alimentación VCC del circuito y de las tensiones de salida de las compuertas. La figura 4.2 muestra los diagramas de flujo de estas pruebas.

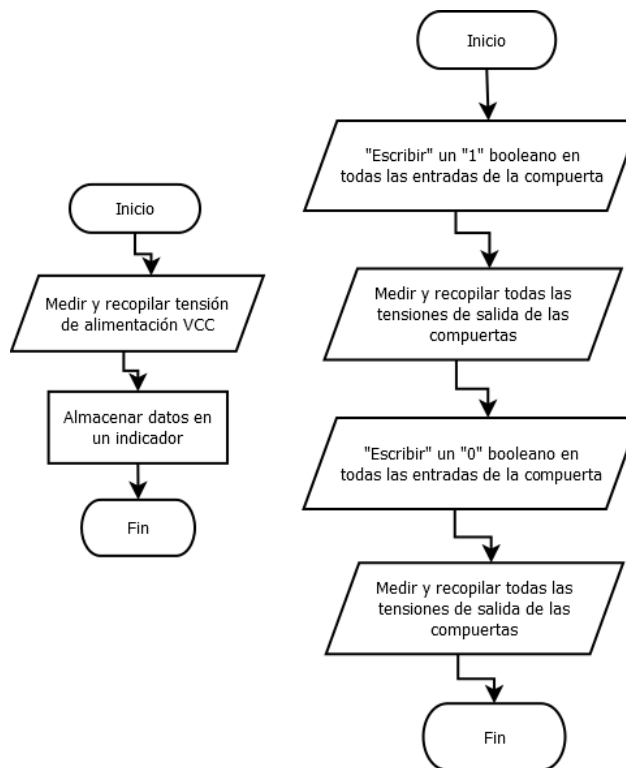


Figura 4.2. Diagrama de flujo de las pruebas eléctricas de la compuerta AND

El diagrama de flujo de la izquierda corresponde al de la prueba AND VCC TEST, en esta por medio del DAQ se midió la tensión de alimentación del circuito y el dato se recopilaba en un indicador numérico como muestra la figura 4.3 del Front Panel de la prueba. Este indicador numérico era el dato que se enviaba a TestStand para validación.

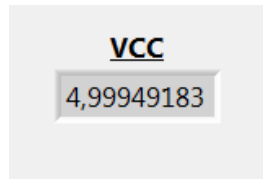


Figura 4.3. Indicador numérico de la prueba AND VCC

El diagrama de flujo de la derecha corresponde a la prueba AND OUTPUT VOLT TEST, que se encargó de medir las tensiones de salida del circuito en las salidas de las compuertas, según fuera su entrada respectiva un 1 o un 0 lógico. Una vez medidas las salidas, los datos se recopilaban en indicadores numéricos según correspondiera su salida y su estado lógico. La figura 4.4 del Front Panel de esta prueba, muestra estos resultados. Estos indicadores numéricos son los datos que se enviaban a TestStand para su validación.

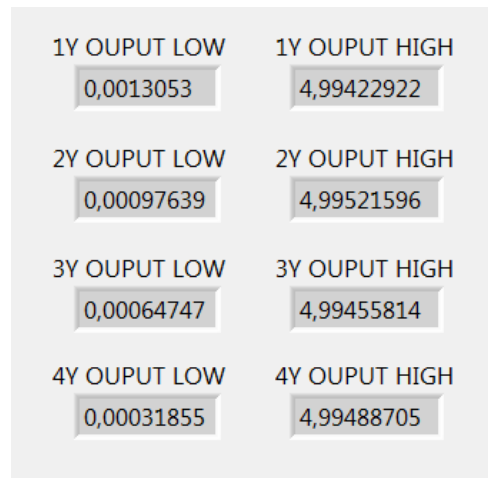


Figura 4.4. Indicadores numéricos de la prueba AND OUTPUT VOLT TEST

Pruebas Funcionales

La prueba de funcionalidad del circuito correspondió en evaluar las tablas de verdad de cada una de las compuertas del chip, a partir de entradas digitales generadas aleatoriamente. La figura 4.5 detalla el procedimiento de esta prueba:

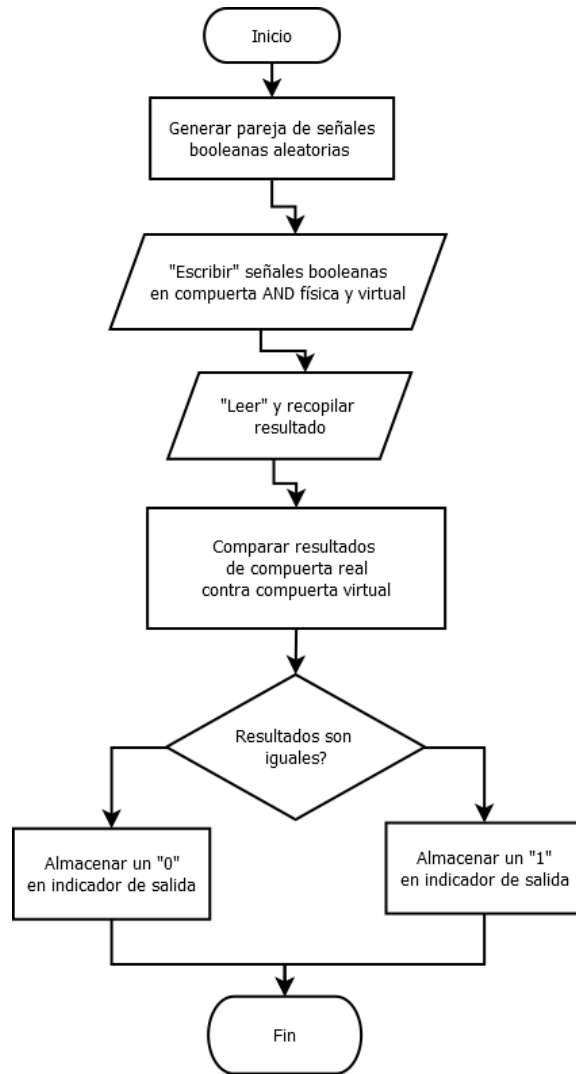


Figura 4.5. Diagrama de flujo de la prueba AND TRUTH TABLE TEST

La figura 4.6 muestra el Front Panel de esta prueba, donde se utilizaban indicadores booleanos para recopilar la combinación de señales digitales en las entradas de cada compuerta, las salidas tanto de la compuerta real como de la virtual y el resultado de la comparación donde si estaba correcto, el indicador sería verde. El indicador numérico del final es el dato que se enviaba a TestStand para la validación de la compuerta.



Figura 4.6. Front Panel de la prueba AND TRUTHTABLE TEST para un resultado correcto

En caso de que la compuerta estuviera defectuosa, el indicador booleano de la comparación se mostraba de color rojo y el número que se almacenaba en el indicador numérico era 0. La figura 4.7 muestra este caso.



Figura 4.7. Front Panel de la prueba AND TRUTHTABLE TEST para un resultado incorrecto

Para más información sobre el diseño de los VI's, consultar el Anexo 0.

Secuencia de pruebas para validación

En este apartado se mostrará el funcionamiento de la secuencia por medio de su diagrama de flujo y se explicarán las reglas utilizadas para evaluar los datos recopilados de cada una de las pruebas. En la figura 4.8 se muestra el diagrama de flujo de la secuencia de la compuerta AND:

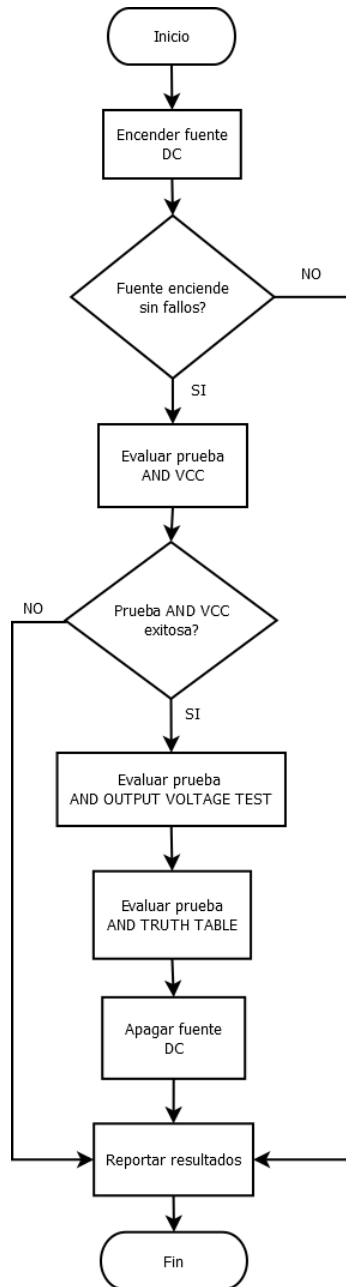


Figura 4.8. Diagrama de flujo de la secuencia de pruebas de la compuerta AND

La figura 4.9 muestra como se ve la secuencia de la compuerta AND en el software TestStand.

| | | | |
|-------------------------|--|--|-------------|
| Setup (1) | | | |
| DC Power ON | Action, NI-DCPower Source Turn On.vi | | Post Action |
| <End Group> | | | |
| Main (5) | | | |
| AND VCC | Numeric Limit Test, 4.75 <= x <= 5.25, AND_VCC - Copy.vi | | Loop |
| if | PASS AND VCC | | |
| AND OUTPUT VOLTAGE TEST | Multiple Numeric Limit Test, AND_OUTPUTVOLT_V2.vi, Numb... | | Loop |
| AND TRUTH TABLE TEST | Multiple Numeric Limit Test, AND_TruthTable_ALT.vi, Number ... | | Loop |
| End | | | |
| <End Group> | | | |
| Cleanup (1) | | | |
| DC Power OFF | Action, NI-DCPower Source Turn Off.vi | | |
| <End Group> | | | |

Figura 4.9. Secuencia de pruebas en TestStand para la compuerta AND

Como muestra la figura 4.10, los parámetros del valor de la tensión de la fuente y la cantidad de iteraciones de cada prueba se controlaban por medio de las variables locales de la herramienta.

| Variables | | | |
|-------------------------------|-------|---------------------------|---------|
| Name | Value | Type | Comment |
| Locals ("MainSequence") | | | |
| ResultList | | Array of Result[0..empty] | |
| Iteraciones | 3 | Number | |
| VCC | 5 | Number | |
| <Right click to insert Local> | | | |

Figura 4.10. Variables locales de TestStand para la secuencia de la compuerta AND

La secuencia iniciaba primero con un VI que se encargaba de encender la fuente DC del PXI y asignarle el voltaje de operación. La acción que se encargaba de llamar este VI tenía una configuración de “Post-Action” donde, en caso de fallo, terminaba la secuencia. La figura 4.11 muestra el reporte de resultados para el caso donde se detectó un fallo en la fuente de alimentación. Como se observa, en el encabezado el resultado del reporte es “Terminated” y en la sección de resultados se muestra la acción de encendido de la fuente como “Failed”, cerrando el reporte al cumplirse la condición de fallo.

UUT Report

| | |
|-------------------|---------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | 1 |
| Date | martes, 20 de noviembre de 2018 |
| Time | 11:02:33 a.m. |
| Operator | administrator |
| Execution Time | 2.43536 seconds |
| Number of Results | 2 |
| UUT Result | Terminated |

Expand / Collapse MainSequence

Begin Sequence: MainSequence
(C:\Users\nipxi\Desktop\Cesar\Pruebas\Secuencias\AND\AND_SEC_RT.seq)

| Step | Status | Measurement | Units | Nom |
|--------------|--------|-------------|-------|-----|
| DC Power ON | Failed | | | |
| DC Power OFF | Done | | | |

End Sequence: MainSequence

End UUT Report

Figura 4.11. Reporte de resultados para el caso de fallo en la fuente DC

Para el módulo AND VCC TEST se escogió una prueba de tipo “Numeric Limit Test”, al ser su entrada un indicador numérico como se observó en la Figura 4.3. Este dato se evaluaba según la regla de comparación “GELE” (Greater Equal Lower Equal), como muestra la figura 4.12 donde se indican también los límites utilizados en la prueba.

| | |
|-----------------|--------------|
| Comparison Type | GELE (>= <=) |
| Low | 4.75 |
| High | 5.25 |
| Units | |
| Numeric Format | <Default> |

Figura 4.12. Regla de comparación para la prueba AND VCC TEST

Como la alimentación influye directamente en los resultados de las siguientes pruebas, se agregó una acción condicional de tipo “If” cuya condición era que, si la prueba AND VCC TEST no era exitosa, no realizaba el resto de las pruebas y terminaba la secuencia. Un caso de fallo de esta prueba se puede observar en la figura 4.13 donde los datos medidos no cumplieron con las reglas definidas y al ser la condición del “If” falsa, no se realizaron el resto de las pruebas.

| Step | Status | Measurement | Units | Limits | | | |
|------------------------|--------|-------------|-------|---------------|-----------|------------|-----------------|
| | | | | Nominal Value | Low Limit | High Limit | Comparison Type |
| DC Power ON | Done | | | | | | |
| AND VCC | Failed | 2.0356496 | | | 4.75 | 5.25 | GELE(>= <=) |
| Number of Loops: | 3 | | | | | | |
| Number of Passes: | 0 | | | | | | |
| Number of Failures: | 3 | | | | | | |
| Final Loop Index: | 3 | | | | | | |
| AND VCC (Loop Indices) | | | | | | | |
| AND VCC (Loop Index:0) | Failed | 2.049464 | | | 4.75 | 5.25 | GELE(>= <=) |
| AND VCC (Loop Index:1) | Failed | 2.0287424 | | | 4.75 | 5.25 | GELE(>= <=) |
| AND VCC (Loop Index:2) | Failed | 2.0356496 | | | 4.75 | 5.25 | GELE(>= <=) |
| If{False} | Done | | | | | | |
| End{if} | Done | | | | | | |
| DC Power OFF | Done | | | | | | |

End Sequence: MainSequence

Figura 4.13. Reporte de resultados para el caso de fallo de la prueba AND VCC TEST

Si el condicional se cumplía, se procedía a la prueba AND OUTPUT VOLTAGE TEST. Este módulo era de tipo “Multiple Numeric Limit Test”, al tener su VI de prueba múltiples indicadores numéricos. En este caso se necesitaron dos reglas diferentes de comparación, la comparación “GE” (Greater Equal) para la salida en “alto” y la comparación “LE” (Lower Equal) para la salida en bajo. La figura 4.14 muestra ambas reglas y como se asignaron para cada uno de los indicadores del VI, además muestra los límites utilizados en la prueba.

| Index | Measurement Name | Settings | | Comparison Type | GE (>=) |
|-------|------------------|----------|---|-----------------|-----------|
| 0 | 1Y Output High | >= 2.4 | + | Low | 2.4 |
| 1 | 2Y Output High | >= 2.4 | - | High | 1 |
| 2 | 3Y Output High | >= 2.4 | ▲ | Units | volt |
| 3 | 4Y Output High | >= 2.4 | ▼ | Numeric Format | <Default> |
| 4 | 1Y Output Low | <= 0.4 | | Comparison Type | LE (<=) |
| 5 | 2Y Output Low | <= 0.4 | | Low | 0.4 |
| 6 | 3Y Output Low | <= 0.4 | | High | 1 |
| 7 | 4Y Output Low | <= 0.4 | | Units | volt |
| | | | | Numeric Format | <Default> |

Figura 4.14. Reglas de comparación para la prueba AND OUTPUT VOLTAGE TEST

La última prueba era la de AND TRUTHTABLE TEST, que correspondía a una prueba de tipo “Multiple Numeric Limit Test”. Esta prueba evaluaba el indicador numérico del VI de esta prueba, comprobando si el resultado era 1 o 0. Debido a esto, la regla de comparación asignada fue “GE” (Greater Equal) como muestra la figura 4.15.

| Index | Measurement Name | Settings | | Comparison Type | GE (>=) |
|-------|------------------|----------|---|-----------------|-----------|
| 0 | Tabla #1 | >= 0.1 | + | Low | 0.1 |
| 1 | Tabla #2 | >= 0.1 | - | High | 11 |
| 2 | Tabla #3 | >= 0.1 | ▲ | Units | |
| 3 | Tabla #4 | >= 0.1 | ▼ | Numeric Format | <Default> |

Figura 4.15. Reglas de comparación para la prueba AND TRUTHTABLE TEST

Finalmente, si todos los módulos de prueba se ejecutaban de manera correcta, se procedía a apagar la fuente DC y enviar los datos para que TestStand construyera el reporte de validación.

Resultados de validación de la compuerta AND

En esta sección se procederá a mostrar los resultados de las mediciones y aplicaciones de las pruebas anteriormente descritas.

Para la validación del entorno diseñado se aplicaron dos grupos de mediciones. En la primera medición se ejecutó la secuencia de TestStand en función de 3 grupos de iteraciones que escalaban en cantidad, esto con el fin de comprobar el comportamiento del entorno al aumentar la cantidad de iteraciones de la secuencia. Además, para este primer grupo de mediciones se utilizaron 3 marcas diferentes de compuertas en función de analizar si el entorno operaba de diferente manera dependiendo de la marca del circuito bajo pruebas.

Las figuras 4.16, 4.17 y 4.18 muestran los encabezados de los reportes de resultados para las compuertas de las marcas Motorola, Texas Instruments y Renesas, respectivamente. Cada figura agrupo los resultados de las diferentes versiones de ejecución de la secuencia, en función de la marca. Para las pruebas se utilizó la misma alimentación de 5V y las reglas de comparación para todas las pruebas fueron definidas en función de los parámetros descritos en la tabla 4.3.

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Motorola_10 Iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:17:13 p.m. |
| Operator | administrator |
| Execution Time | 5.45062 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Motorola_20 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:55:28 p.m. |
| Operator | administrator |
| Execution Time | 7.67875 seconds |
| Number of Results | 67 |
| UUT Result | Passed |

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Motorola_30 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:51:21 p.m. |
| Operator | administrator |
| Execution Time | 10.8305 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.16. Resumen del reporte de resultados de la compuerta marca Motorola para los 3 grupos de iteraciones

| | |
|-------------------|----------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Texas Instruments_10 Iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:16:55 p.m. |
| Operator | administrator |
| Execution Time | 5.75835 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

| | |
|-------------------|----------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Texas Instruments_20 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:54:59 p.m. |
| Operator | administrator |
| Execution Time | 7.99419 seconds |
| Number of Results | 67 |
| UUT Result | Passed |

| | |
|-------------------|----------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Texas Instruments_30 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:50:56 p.m. |
| Operator | administrator |
| Execution Time | 11.128 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.17. Resumen del reporte de resultados de la compuerta marca TI para los 3 grupos de iteraciones

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Renesas_10 Iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:17:26 p.m. |
| Operator | administrator |
| Execution Time | 5.60745 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Renesas_20 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:55:42 p.m. |
| Operator | administrator |
| Execution Time | 7.6184 seconds |
| Number of Results | 67 |
| UUT Result | Passed |

| | |
|-------------------|--------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Renesas_30 iteraciones |
| Date | lunes, 19 de noviembre de 2018 |
| Time | 06:51:43 p.m. |
| Operator | administrator |
| Execution Time | 10.8555 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.18. Resumen del reporte de resultados de la compuerta marca Renesas para los 3 grupos de iteraciones

Los resultados de estas pruebas se resumen en la tabla 4.5.

Tabla 4.5. Resultados de la secuencia AND en función de la cantidad de iteraciones

| Marca | Cantidad de Iteraciones | Duración (s) | Porcentaje de éxito (%) |
|-------------------|-------------------------|--------------|-------------------------|
| Motorola | 10 | 5.45062 | 100 |
| | 20 | 7.67875 | 100 |
| | 30 | 10.8305 | 100 |
| Texas Instruments | 10 | 5.75835 | 100 |
| | 20 | 7.99419 | 100 |
| | 30 | 11.128 | 100 |
| Renesas | 10 | 5.60745 | 100 |
| | 20 | 7.6184 | 100 |
| | 30 | 10.8555 | 100 |

De los resultados anteriormente expuestos, se observó que el entorno puede aumentar la cantidad de iteraciones de las pruebas sin que cambie el resultado positivo de esta, más allá de los tiempos de duración. Entre grupos de iteración, el tiempo aumentaba aproximadamente en 2.689 segundos, 2.684 segundos y 2.624 segundos para las 3 marcas respectivamente. Se observa con esto que independientemente de la marca, el comportamiento del entorno no cambia y varía los tiempos de manera similar entre las marcas.

El segundo grupo de mediciones consistió en aplicar el entorno de validación en una compuerta AND catalogada como dañada, con el fin de analizar el comportamiento del entorno al detectar fallos en el circuito bajo prueba.

Para la prueba se utilizó una compuerta de la marca Texas Instruments defectuosa, las condiciones de ejecución fueron una alimentación VCC de 5V y un ciclo de 30 iteraciones, para la comparación de los valores eléctricos se utilizaron los datos de la tabla 4.3. A continuación, se presentan imágenes de extractos del reporte de la secuencia, donde se muestran los resultados de cada prueba y el análisis de dichos datos para identificar el fallo de la compuerta:

| | | | | | | | | |
|--------------|--------|-----------|------|--|--|-----|-----|-------------|
| DC Power ON | Done | | | | | | | |
| AND VCC | Failed | | | | | | | |
| Measurement: | | | | | | | | |
| VCC | Failed | 2.0695278 | volt | | | 4.5 | 5.5 | GELE(>= <=) |
| VCC | Failed | 2.0642652 | volt | | | 4.5 | 5.5 | GELE(>= <=) |
| VCC | Failed | 2.0682122 | volt | | | 4.5 | 5.5 | GELE(>= <=) |
| VCC | Failed | 2.0662387 | volt | | | 4.5 | 5.5 | GELE(>= <=) |
| VCC | Failed | 2.0698567 | volt | | | 4.5 | 5.5 | GELE(>= <=) |

Figura 4.19. Reporte de resultados de la prueba AND VCC TEST

Al ejecutar la secuencia en la compuerta defectuosa, TestStand detectó un fallo en una de las comprobaciones, la prueba de alimentación AND VCC TEST. Según el reporte de resultados, la compuerta entregaba una medición promedio de 2.06V. muy por debajo del mínimo requerido por el fabricante. Al no tener una alimentación correcta, la secuencia se detenía y no realizaba el resto de las evaluaciones. Para tener un análisis más completo del comportamiento, se modificó la secuencia para que permitiera realizar el resto de las pruebas y así tener más resultados para el análisis.

| Step | Status | Measurement | Units | Limits | | | |
|-------------------------|--------|-------------|-------|---------------|-----------|------------|-----------------|
| | | | | Nominal Value | Low Limit | High Limit | Comparison Type |
| AND OUTPUT VOLTAGE TEST | Failed | | | | | | |
| Measurement: | | | | | | | |
| 1Y Output High | Failed | 1.5695768 | volt | | 2.4 | | GE(>=) |
| 2Y Output High | Failed | 0.65946702 | volt | | 2.4 | | GE(>=) |
| 3Y Output High | Failed | 0.83379252 | volt | | 2.4 | | GE(>=) |
| 4Y Output High | Failed | 1.4656396 | volt | | 2.4 | | GE(>=) |
| 1Y Output Low | Passed | 0.013146302 | volt | | 0.4 | | LE(<=) |
| 2Y Output Low | Passed | 0.012817386 | volt | | 0.4 | | LE(<=) |
| 3Y Output Low | Passed | 0.12530687 | volt | | 0.4 | | LE(<=) |
| 4Y Output Low | Passed | 0.011501719 | volt | | 0.4 | | LE(<=) |
| Number of Loops: | 30 | | | | | | |
| Number of Passes: | 0 | | | | | | |
| Number of Failures: | 30 | | | | | | |

Figura 4.20. Reporte de resultados de la prueba AND OUTPUT VOLTAGE TEST

Al ejecutarse la segunda prueba de las tensiones de salida, se constató que todas las salidas en “alto” daban valores muy por debajo del mínimo requerido, entre 0.6 y 1.5V, como no son valores de 0 como las salidas en “bajo” que rondan los 0.012V, se puede considerar estos resultados como un tercer estado no acorde al funcionamiento de la compuerta.

| | | | | | | | |
|----------------------|--------|---|--|--|-----|--|--------|
| AND TRUTH TABLE TEST | Failed | | | | | | |
| Measurement: | | | | | | | |
| Tabla #1 | Failed | 0 | | | 0.1 | | GE(>=) |
| Tabla #2 | Failed | 0 | | | 0.1 | | GE(>=) |
| Tabla #3 | Failed | 0 | | | 0.1 | | GE(>=) |
| Tabla #4 | Failed | 0 | | | 0.1 | | GE(>=) |
| Number of Loops: | 30 | | | | | | |
| Number of Passes: | 0 | | | | | | |
| Number of Failures: | 30 | | | | | | |

Figura 4.21. Reporte de resultados de la prueba AND TRUTHTABLE TEST

La última prueba fue la de funcionalidad de la compuerta donde el reporte indicaba que, en todas las iteraciones ejecutadas, se incumplió con la tabla de verdad de la compuerta. Este resultado está directamente relacionado con el hecho de que las salidas en alto no estaban dando el valor de tensión mínimo requerido (2.4V) para ser detectados como un 1 booleano, como se observó en los resultados de la figura 4.20. Este resultado incorrecto se detectó en las 4 compuertas que componen el chip.

Basándose en los resultados antes descritos se llegó a la conclusión que el fallo de la compuerta se debía a un posible cortocircuito en la alimentación VCC de la compuerta, al no existir ningún fallo con la fuente de alimentación del PXI por lo que al dispositivo si se le suministró la alimentación requerida (5V). Al existir este cortocircuito en la alimentación,

ninguna de las salidas en “alto” alcanzaron el mínimo requerido de 2.4V provocando que tampoco se cumpliera con las tablas de verdad.

4.3 Prueba de Concepto #2: Memoria SRAM 2114

Parámetros Eléctricos y Funcionales del Circuito

Para esta prueba de concepto se seleccionó la memoria SRAM 2114, con el fin de probar el comportamiento del entorno ante un circuito que combina características secuenciales y combinacionales. La hoja de datos de la memoria [16] brindó los detalles del mapeo de pines del chip, sus parámetros eléctricos y funcionales, y sus gráficos de tiempos. A continuación, la figura 4.22 muestra la distribución de pines de la memoria.

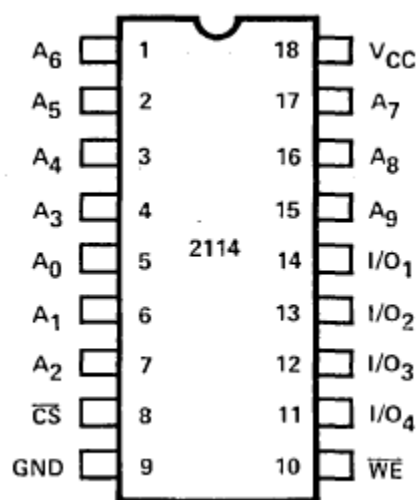


Figura 4.22. Diagrama de pines de la memoria 2114 [16]

La tabla 4.6 muestra los parámetros eléctricos que se seleccionaron para ser validados en el entorno.

Tabla 4.6. Tabla de datos eléctricos de la memoria 2114 [16]

| Parámetro | Valor Mínimo | Valor Típico | Valor Máximo | Unidad |
|-----------|--------------|--------------|--------------|--------|
| VCC | 4.5 | 5 | 5.5 | V |
| VIH | 2 | - | VCC | V |
| VIL | -0.5 | - | 0.8 | V |
| VOH | 2.4 | - | VCC | V |
| VOL | - | - | 0.4 | V |

La tabla 4.7 muestra la configuración combinacional para ejecutar las tareas de escritura y lectura de la memoria.

Tabla 4.7. Tabla de verdad de las funciones de la memoria 2114 [16]

| CS | WE | DQ | Función |
|----|----|------|-------------------|
| 1 | X | Z | Chip inhabilitado |
| 0 | 0 | 1 | Escribir "1" |
| 0 | 0 | 0 | Escribir "0" |
| 0 | 1 | DATO | Leer Dato |

Pruebas Eléctricas

Para la validación de la memoria SRAM, se diseñaron 2 pruebas eléctricas para validar la alimentación VCC del circuito y las tensiones de salida de los pines de datos I/O. La figura 4.23 muestra los diagramas de flujo de estas pruebas.

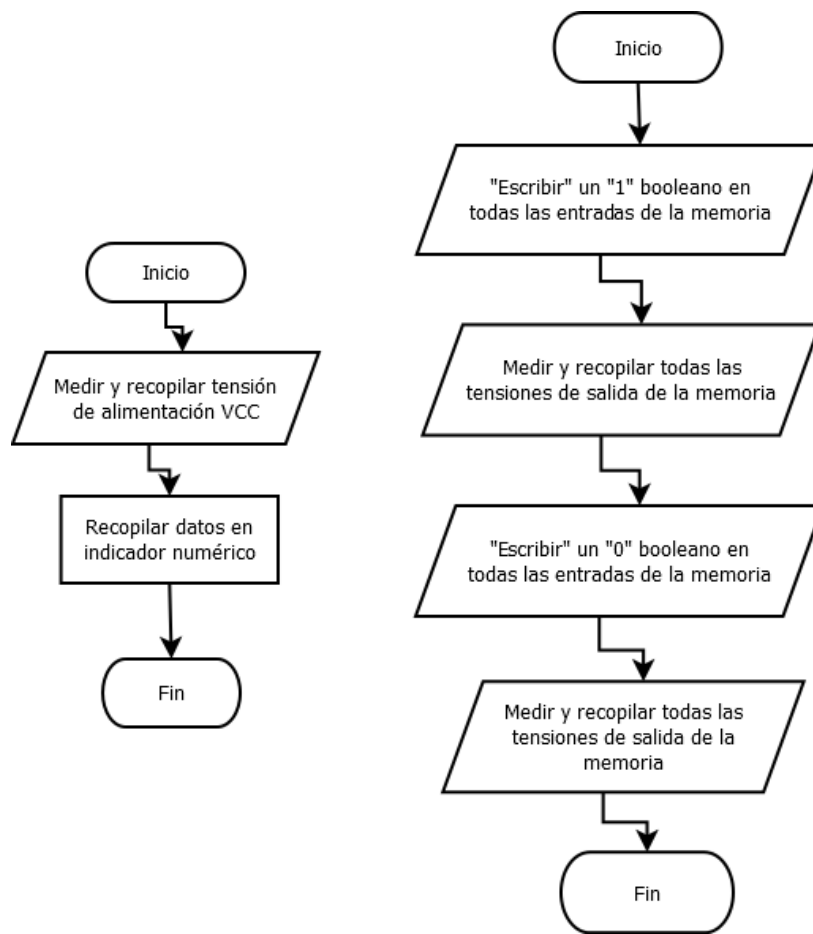


Figura 4.23. Diagramas de flujo de las pruebas eléctricas de la memoria SRAM

El diagrama de flujo de la izquierda muestra el diseño de la prueba SRAM VCC TEST donde, por medio del DAQ del PXI, se midió la tensión de alimentación de la memoria y se recopilaba en un indicador numérico como se observa en la figura 4.24.

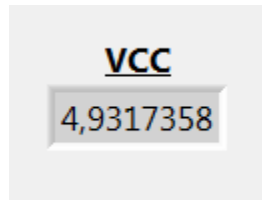


Figura 4.24. Front panel de la prueba SRAM VCC TEST

El diagrama de flujo de la derecha corresponde a la prueba SRAM OUTPUT VOLT TEST donde se medían las tensiones de salida de los pines de datos I/O de la memoria. Las mediciones se recopilaban en indicadores numéricos como lo muestra la figura 4.25.

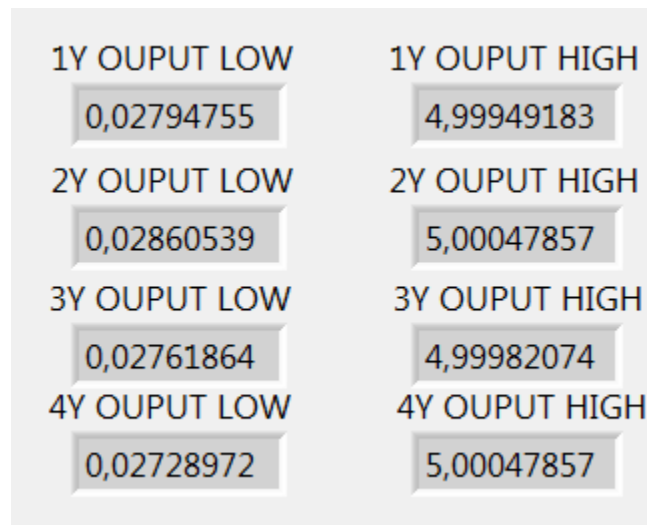


Figura 4.25. Front Panel de la prueba SRAM OUTPUT VOLT TEST

Pruebas Funcionales

La prueba de funcionalidad de la SRAM consistió en escribir el mismo dato aleatorio en todos los espacios de memoria, posteriormente leer todos los espacios de memoria y comparar lo que se recopiló de memoria con el dato originalmente escrito. La figura 4.26 muestra el diagrama de flujo de esta prueba.

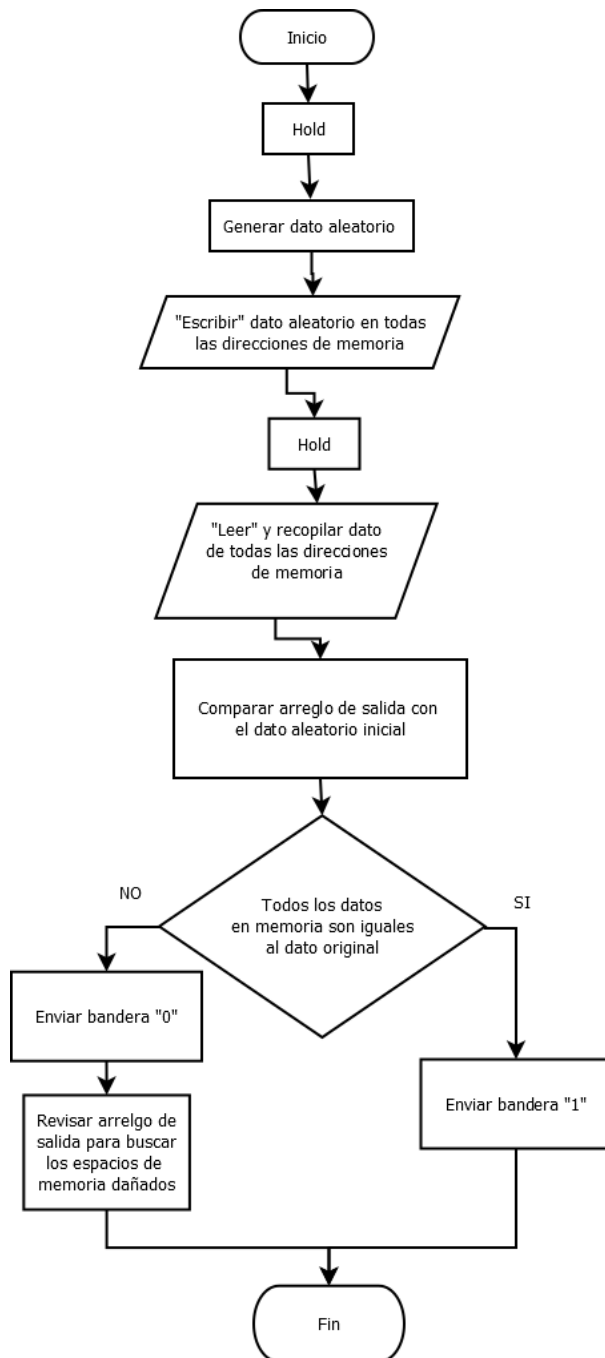


Figura 4.26. Diagrama funcional de la prueba SRAM MEMORY TEST

La figura 4.27 presenta el Front Panel de esta prueba, donde los indicadores muestran el dato que se escribió en todos los espacios de dirección, la dirección de memoria donde se escribió, los estados de la memoria, los pines de control de la memoria, el resultado de la comparación y en caso de fallo, la cantidad de errores y la ubicación del espacio de memoria donde se detectó el fallo.

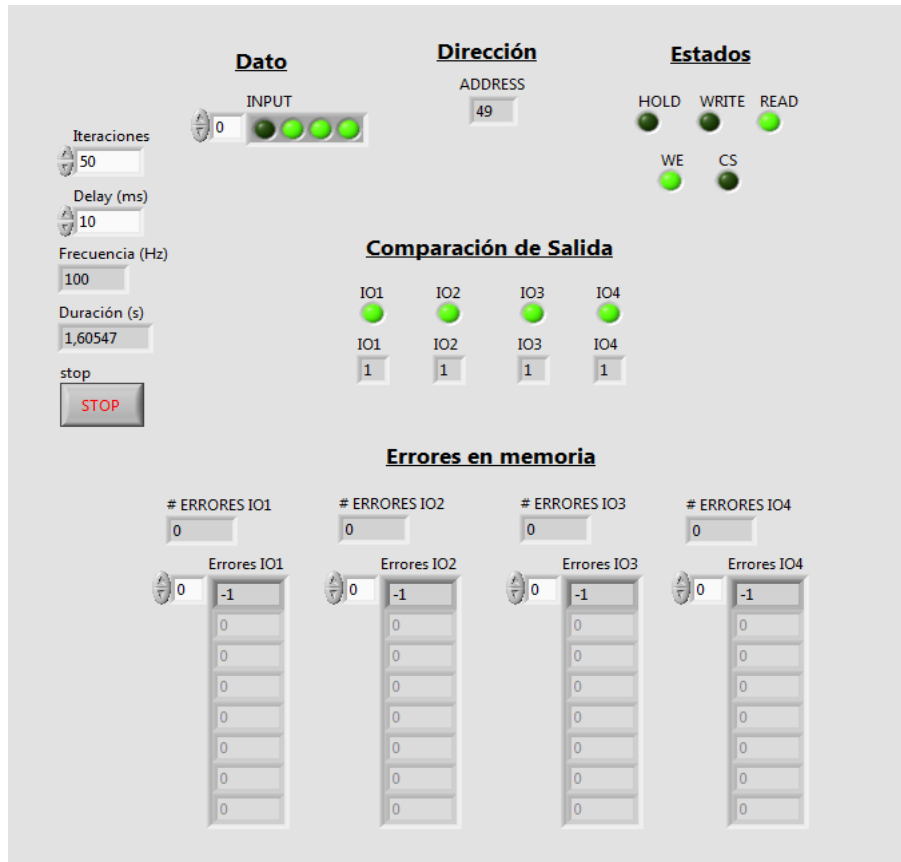


Figura 4.27. Front panel de la prueba SRAM MEMORY TEST para el caso correcto

La figura 4.28 muestra el mismo Front Panel anterior, pero para el caso en el que se detectó un fallo en la memoria. Para este caso El indicador IO se vuelve de color rojo si en alguno de los pines de memoria se encontró una incongruencia entre lo escrito y lo leído. En la tabla Errores IO4 se muestra las ubicaciones de los espacios de memoria donde se detectaron las incongruencias.



Figura 4.28. Front panel de la prueba SRAM MEMORY TEST para el caso incorrecto

Secuencia de pruebas para validación

En esta sección se explicará el diseño de la secuencia de pruebas para la memoria por medio de su diagrama de flujo y se mostrarán las configuraciones de las reglas de comparación de cada prueba. En la figura 4.29 se muestra el diagrama de flujo de la secuencia de pruebas de la memoria SRAM.

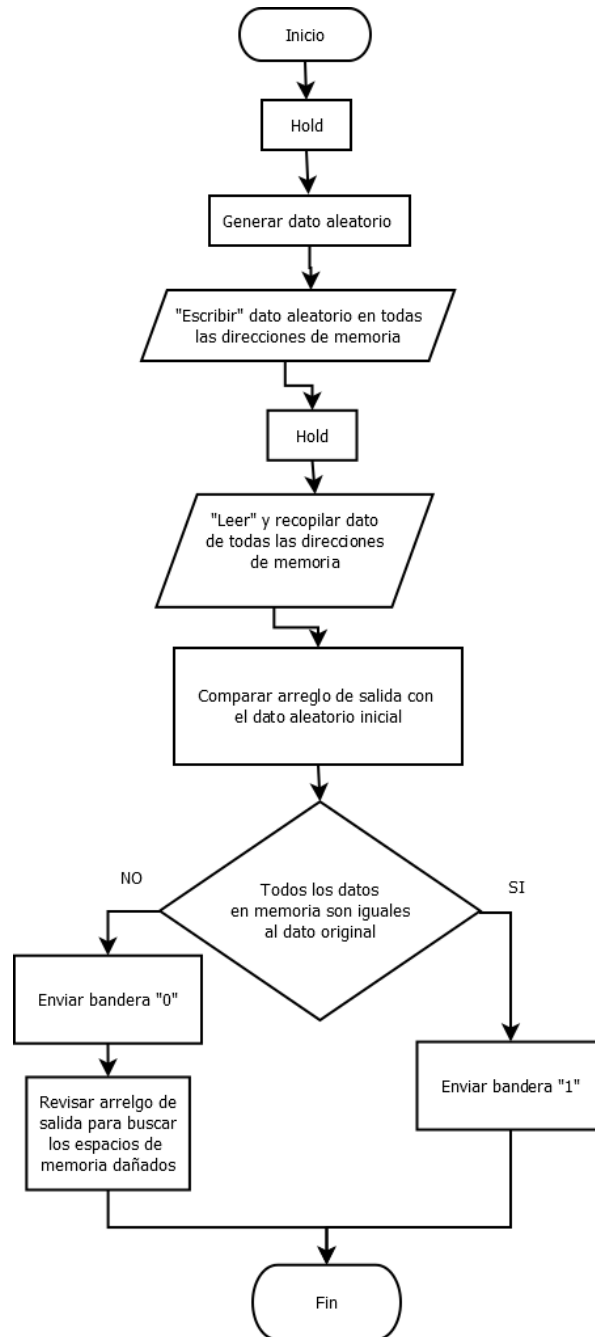


Figura 4.29. Diagrama de flujo de la secuencia de pruebas de la memoria SRAM

La figura 4.30 muestra como se ve la secuencia de pruebas de la memoria en la herramienta TestStand.

| Step | Description | Settings |
|--------------------------|---|-------------|
| Setup (1) | | |
| DC Power ON | Action, NI-DCPower Source Turn On.vi | Post Action |
| <End Group> | | |
| Main (5) | | |
| SRAM VCC | Numeric Limit Test, 4.5 <= x <= 5.5, SRAM_VC... | Loop |
| If | PASS SRAM VCC | |
| SRAM OUTPUT VOLTAGE TEST | Multiple Numeric Limit Test, SRAM_OUTPUTV... | Loop |
| SRAM MEMORY TEST | Multiple Numeric Limit Test, SRAM TEST_V4.v... | Loop |
| End | | |
| <End Group> | | |
| Cleanup (1) | | |
| DC Power OFF | Action, NI-DCPower Source Turn Off.vi | |
| <End Group> | | |

Figura 4.30. Secuencia de pruebas en TestStand para la memoria SRAM

Para esta secuencia de pruebas, se manejaron las variables locales de iteraciones, VCC, periodo y cantidad de direcciones. La figura 4.31 muestra la tabla de estas variables en TestStand.

| Variables | | |
|-------------------------|-------|---------------------------|
| Name | Value | Type |
| Locals ('MainSequence') | | |
| ResultList | | Array of Result[0..empty] |
| Iteraciones | 30 | Number |
| VCC | 5 | Number |
| Periodo | 1 | Number |
| Cant_direcc | 1024 | Number |

Figura 4.31. Variables locales de TestStand para la secuencia de la memoria SRAM

La estructura de esta secuencia de pruebas es igual a la estructura de la secuencia de la compuerta AND, se inicia con el VI que activa la fuente DC del PXI y se continua con las pruebas eléctricas y funcionales. Los casos de error en la secuencia son los mismo que se asignaron en la compuerta AND, explícitamente, los casos donde ocurre un fallo de la fuente o un fallo en la alimentación VCC.

La figura 4.32 muestra el reporte de la secuencia para el caso de fallo de la fuente DC.

UUT Report

| | |
|-------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | BAD |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:16:26 a.m. |
| Operator | administrator |
| Execution Time | 2.33337 seconds |
| Number of Results | 2 |
| UUT Result | Terminated |

Expand / Collapse MainSequence

Begin Sequence: MainSequence
(C:\Users\nipxi\Desktop\Cesar\Pruebas\Secuencias\SRAM\SRAM_SEC_ALT.seq)

| Step | Status | Measurement | Units | Nom |
|--------------|--------|-------------|-------|-----|
| DC Power ON | Failed | | | |
| DC Power OFF | Done | | | |

End Sequence: MainSequence

End UUT Report

Figura 4.32. Reporte de resultados para el caso de fallo en la fuente DC

La figura 4.33 muestra la regla de comparación utilizada en la prueba SRAM VCC TEST y los límites asignados. Esta prueba se configuró para que fuera de tipo “Numeric Limit Test”

| | |
|-----------------|--------------|
| Comparison Type | GELE (>= <=) |
| Low | 4.5 |
| High | 5.5 |
| Units | |
| Numeric Format | <Default> |

Figura 4.33. Regla de comparación de la prueba SRAM VCC TEST

Esta prueba también estaba configurada, al igual que con la compuerta, para que en caso de que la alimentación VCC no fuera la correcta no se ejecutaran el resto de las pruebas. La figura 4.34 muestra el reporte de resultados de la secuencia de pruebas de la memoria, para el caso de fallo en la alimentación VCC.

| Step | Status | Measurement | Units | Limits | | | |
|-------------------------|--------|-------------|-------|---------------|-----------|------------|-----------------|
| | | | | Nominal Value | Low Limit | High Limit | Comparison Type |
| DC Power ON | Done | | | | | | |
| SRAM VCC | Failed | 1.7238381 | | | 4.5 | 5.5 | GELE(>= <=) |
| Number of Loops: | 5 | | | | | | |
| Number of Passes: | 0 | | | | | | |
| Number of Failures: | 5 | | | | | | |
| Final Loop Index: | 5 | | | | | | |
| SRAM VCC (Loop Indices) | | | | | | | |
| SRAM VCC (Loop Index:0) | Failed | 1.7231803 | | | 4.5 | 5.5 | GELE(>= <=) |
| SRAM VCC (Loop Index:1) | Failed | 1.7228513 | | | 4.5 | 5.5 | GELE(>= <=) |
| SRAM VCC (Loop Index:2) | Failed | 1.7231803 | | | 4.5 | 5.5 | GELE(>= <=) |
| SRAM VCC (Loop Index:3) | Failed | 1.7238381 | | | 4.5 | 5.5 | GELE(>= <=) |
| SRAM VCC (Loop Index:4) | Failed | 1.7238381 | | | 4.5 | 5.5 | GELE(>= <=) |
| If{False} | Done | | | | | | |
| End{If} | Done | | | | | | |
| DC Power OFF | Done | | | | | | |

End Sequence: MainSequence

End UUT Report

Figura 4.34. Reporte de resultados para el caso de fallo de la prueba SEAM VCC TEST

La siguiente prueba, SRAM OUTPUT VOLT TEST, era de tipo “Multiple Numeric Limit Test” y se usaron dos reglas de comparación para las tensiones de salida. Se usó la comparación “GE” (Greater Equal) para la salida en “alto” y “LE” (Lower Equal) para la salida en “bajo”. La figura 4.35 muestra la tabla de estas reglas en la herramienta TestStand

| Index | Measurement Name | Settings | | Comparison Type | GE (>=) |
|-------|------------------|----------|---|-----------------|-----------|
| 0 | 1Y Output High | >= 2.4 | + | Low | 2.4 |
| 1 | 2Y Output High | >= 2.4 | - | High | 1 |
| 2 | 3Y Output High | >= 2.4 | ▲ | Units | volt |
| 3 | 4Y Output High | >= 2.4 | ▼ | Numeric Format | <Default> |
| 4 | 1Y Output Low | <= 0.4 | | Comparison Type | LE (<=) |
| 5 | 2Y Output Low | <= 0.4 | | Low | 0.4 |
| 6 | 3Y Output Low | <= 0.4 | | High | 1 |
| 7 | 4Y Output Low | <= 0.4 | | Units | volt |
| | | | | Numeric Format | <Default> |

Figura 4.35. Reglas de comparación para la prueba SRAM OUTPUT VOLT TEST

El último módulo era el de SRAM MEMORY TEST, que correspondía a una prueba de tipo “Multiple Numeric Limit Test”. Las salidas a evaluar eran los indicadores numéricos de cada uno de los pines I/O de la memoria, comprobando si el resultado era 1 o 0. La regla de comparación era “GE” (Greater Equal). Si el valor era 1, se escribió de manera correcta en la memoria, en caso contrario el valor sería 0. La figura 4.36 muestra la tabla de reglas de esta prueba.

| Index | Measurement Name | Settings | | Comparison Type | GE (>=) |
|-------|------------------|----------|---|-----------------|-----------|
| 0 | IO1 | >= 0.1 | + | Low | 0.1 |
| 1 | IO2 | >= 0.1 | - | High | 11 |
| 2 | IO3 | >= 0.1 | ▲ | Units | |
| 3 | IO4 | >= 0.1 | ▼ | Numeric Format | <Default> |

Figura 4.36. Reglas de comparación de la prueba SRAM MEMORY TEST

Finalmente, si todos los módulos de prueba se ejecutaban de manera correcta, se procedía a apagar la fuente DC y enviar los datos para que TestStand construyera el reporte de validación.

Resultados de validación de la memoria SRAM

En esta sección se procederá a mostrar los resultados de las diferentes mediciones y aplicaciones de las pruebas anteriormente descritas.

Para la validación de este entorno se aplicaron 3 grupos de mediciones. El primer grupo consistió en evaluar el comportamiento del entorno al aumentar la cantidad de iteraciones de las pruebas de la secuencia. Además, en función de comprobar la ejecución del entorno en diferentes marcas, se aplicaron estos incrementos de iteraciones en 3 memorias de diferentes marcas.

Las figuras 4.37, 4.38 y 4.39 muestran los encabezados de los reportes de resultados para las compuertas de las marcas National Semiconductor, NEC Electronics y Synertec, respectivamente. Cada figura agrupo los resultados de las diferentes versiones de ejecución de la secuencia, en función de la marca. Para las pruebas se utilizó la misma alimentación de 5V y una velocidad de reloj de 1kHz. Las reglas de comparación para todas las pruebas fueron definidas en función de los parámetros descritos en la tabla 4.6.

| | |
|-------------------|---------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | National Semiconductor_10 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:29:11 a.m. |
| Operator | administrator |
| Execution Time | 70.687 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

| | |
|-------------------|---------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | National Semiconductor_20 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:48:52 a.m. |
| Operator | administrator |
| Execution Time | 136.458 seconds |
| Number of Results | 67 |
| UUT Result | Passed |

| | |
|-------------------|---------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | National Semiconductor_30 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:54:42 a.m. |
| Operator | administrator |
| Execution Time | 204.79 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.37. Resumen del reporte de resultados de la memoria marca National Semiconductor para los 3 grupos de iteraciones

| | |
|-------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | NEC Electronics_10 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:33:47 a.m. |
| Operator | administrator |
| Execution Time | 70.6632 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

| | |
|-------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | NEC Electronics_20 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:45:51 a.m. |
| Operator | administrator |
| Execution Time | 136.68 seconds |
| Number of Results | 67 |
| UUT Result | Passed |

| | |
|-------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | NEC Electronics_30 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 12:11:31 p.m. |
| Operator | administrator |
| Execution Time | 204.258 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.38. Resumen del reporte de resultados de la memoria marca NEC Electronics para los 3 grupos de iteraciones

| | |
|-------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Synertek_10 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:36:19 a.m. |
| Operator | administrator |
| Execution Time | 71.0699 seconds |
| Number of Results | 37 |
| UUT Result | Passed |
| Station ID | PXI-PA511 |
| Serial Number | Synertek_20 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 11:40:37 a.m. |
| Operator | administrator |
| Execution Time | 135.991 seconds |
| Number of Results | 67 |
| UUT Result | Passed |
| Station ID | PXI-PA511 |
| Serial Number | Synertek_30 iteraciones |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 12:04:45 p.m. |
| Operator | administrator |
| Execution Time | 203.755 seconds |
| Number of Results | 97 |
| UUT Result | Passed |

Figura 4.39. Resumen del reporte de resultados de la memoria marca Synertek para los 3 grupos de iteraciones

Los resultados de estas pruebas se resumen en la tabla 4.8.

Tabla 4.8. Resultados de secuencia SRAM según cantidad de iteraciones

| Marca | Cantidad de Iteraciones | Duración (s) | Porcentaje de éxito (%) |
|------------------------|-------------------------|--------------|-------------------------|
| National Semiconductor | 10 | 70.687 | 100 |
| | 20 | 136.458 | 100 |
| | 30 | 207.79 | 100 |
| NEC Electronics | 10 | 70.6632 | 100 |
| | 20 | 136.68 | 100 |
| | 30 | 204.258 | 100 |
| Synertek | 10 | 71.0699 | 100 |
| | 20 | 135.991 | 100 |
| | 30 | 203.755 | 100 |

De los resultados anteriormente expuestos se observó que, en comparación con la validación de la compuerta AND, los tiempos de ejecución aumentaron entre un 1200% y un 1800% de los tiempos de ejecución de la compuerta AND. Este aumento es entendible al tener la memoria más tareas que la compuerta y se toma en cuenta también que son 1024 espacios de memoria para 4 bits de datos los que se deben procesar. A pesar de que las pruebas tuvieron una duración más extensa, el entorno no presentó problemas al incrementar la cantidad de iteraciones en cada caso y todas las memorias pasaron la validación de forma correcta.

El segundo grupo de mediciones consistió en evaluar el comportamiento del entorno ante los cambios de la velocidad de reloj para un circuito con parte secuencial. En esta prueba se hicieron 3 recorridos de la secuencia con 3 velocidades de reloj diferentes y se recopilaron sus duraciones. Se utilizó la memoria marca Synertek, las condiciones de trabajo fueron una alimentación de 5V, 10 iteraciones para el ciclo de la secuencia y las condiciones de evaluación de la tabla 4.6.

La figura 4.40 muestra los encabezados de los reportes para las pruebas con velocidad de reloj de 1000Hz, 500Hz y 250Hz respectivamente.

| | |
|--------------------------|------------------------------------|
| Station ID | PXI-PA511 |
| Serial Number | Synertek_1000Hz |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 02:20:30 p.m. |
| Operator | administrator |
| Execution Time | 70.9519 seconds |
| Number of Results | 37 |
| UUT Result | Passed |
| Station ID | PXI-PA511 |
| Serial Number | Synertek_500Hz |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 03:02:17 p.m. |
| Operator | administrator |
| Execution Time | 71.6728 seconds |
| Number of Results | 37 |
| UUT Result | Passed |
| Station ID | PXI-PA511 |
| Serial Number | Synertek_250Hz |
| Date | miércoles, 21 de noviembre de 2018 |
| Time | 03:04:59 p.m. |
| Operator | administrator |
| Execution Time | 131.079 seconds |
| Number of Results | 37 |
| UUT Result | Passed |

Figura 4.40. Resumen del reporte de resultados para el caso de medición en función de la velocidad de reloj

La tabla 4.9 resume los resultados de los reportes.

Tabla 4.9. Resultados de secuencia SRAM según frecuencia de reloj

| Marca | Frecuencia (Hz) | Duración (s) | Porcentaje de éxito (%) |
|----------|-----------------|--------------|-------------------------|
| Synertek | 1000 | 70.9519 | 100 |
| | 500 | 71.6728 | 100 |
| | 250 | 131.079 | 100 |

A partir de estas pruebas, se pudo observar que entre las frecuencias de 1000Hz y 500Hz la diferencia de tiempo es de fracciones de segundo, mientras que la prueba ejecutada a 250Hz registró un incremento de casi el doble de su tiempo de ejecución. A pesar de este incremento abrupto en el tiempo de ejecución, los resultados de las pruebas no se vieron afectados negativamente.

Finalmente, se propuso utilizar un modelo de regresión lineal simple para predecir la duración de la secuencia de pruebas de la SRAM según la cantidad de iteraciones. Cabe destacar que esto no es un caso de pruebas vinculante al funcionamiento del entorno, se realizó con el fin de tener una métrica de estimación de lo que puede durar una prueba en función de la cantidad de iteraciones.

Para determinar el modelo, se realizaron varias ejecuciones de la secuencia con diferente cantidad de iteraciones. Para la prueba se usaron las condiciones de alimentación de 5V y una frecuencia de reloj de 1kHz. La tabla 4.10 presenta las mediciones de cada ejecución.

Tabla 4.10. Predicción de duración de secuencia según iteraciones

| Marca | Cantidad de Iteraciones | Duración (s) |
|----------|-------------------------|--------------|
| Synertek | 5 | 37,0943 |
| | 10 | 71,9519 |
| | 15 | 101,154 |
| | 20 | 135,991 |
| | 25 | 167,132 |
| | 30 | 203,755 |

La figura 4.41 muestra el gráfico del modelo y la ecuación de línea con su coeficiente de determinación:

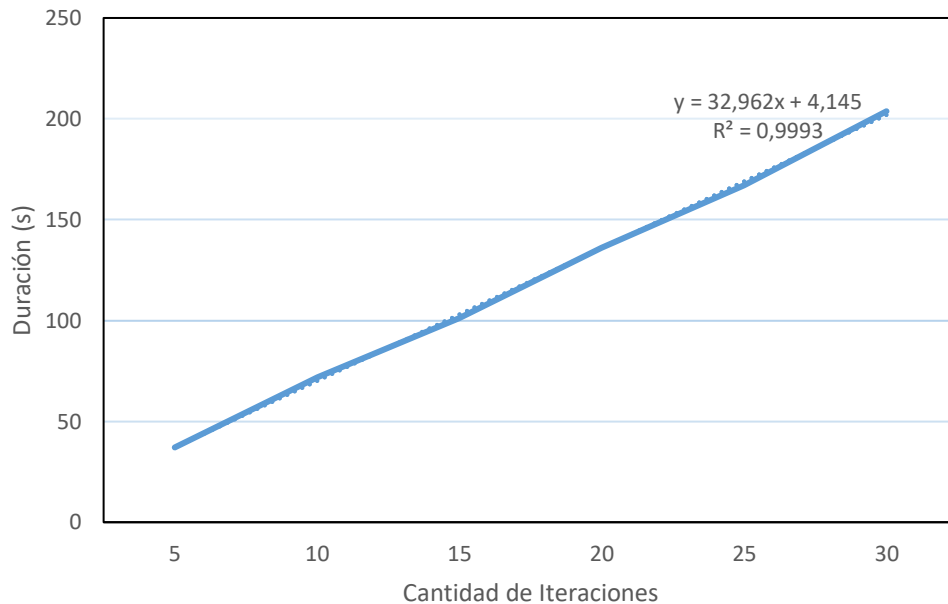


Figura 4.41. Gráfico de duración de la secuencia SRAM en función de la cantidad de iteraciones

Como se observa en el gráfico de línea, el valor R^2 correspondiente al coeficiente de determinación del modelo es de 0,9993, lo que indica que existe una alta relación de comportamiento entre las variables de duración y cantidad de iteraciones al ser un valor muy cercano a 1. Entre más cercano sea el valor del coeficiente de determinación a 1, más fiable es el modelo. Por medio de este modelo se pudo calcular que, para una cantidad de iteraciones de 1000 por prueba, la secuencia duraría 3300.345 ± 2 segundos.

4.4 Propuesta de entorno de validación para procesador RISC-V

En esta sección se explicará la una propuesta de un entorno de validación para el futuro procesador RISC-V del laboratorio DCI-Lab. En relación con que el procesador aún se encuentra en su fase de diseño, la propuesta de entorno de validación post-silicio estará enfocada en el diseño de los módulos que validan las características funcionales del procesador.

Determinación de parámetros eléctricos y funcionales

Como se mencionó antes, el procesador todavía se encuentra en su fase de diseño por lo que no se cuenta con ningún parámetro eléctrico o información referente a las características eléctricas del procesador.

Para el apartado funcional, de la fase de diseño del procesador se tiene completada una simulación de “TOP-LEVEL”, la cual está aplicada en una FPGA Nexys 4. Para validar este “TOP-LEVEL” en la FPGA, los diseñadores del DCI-Lab crearon un bloque de comunicación para pruebas, que permite conectar agentes externos con el CORE simulado. La función de este bloque es permitir ingresar bancos de instrucciones que exciten los bloques internos del CORE, probando así las características funcionales. Estos bancos de instrucciones son secuencias de instrucciones MIPS que desarrollaron los diseñadores del DCI-Lab y que utilizaron para probar la funcionalidad del RTL del CORE. La figura 4.42 muestra cómo está actualmente distribuido el bloque de comunicación de esta simulación:

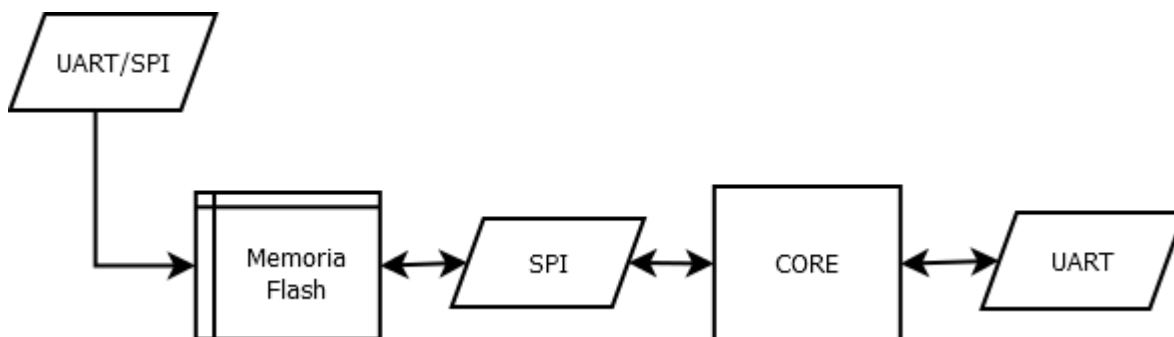


Figura 4.42. Diagrama del bloque de comunicación para pruebas del TOP-LEVEL

De primero se tiene una interfaz UART/SPI donde el primero se encarga de recibir los arreglos de bits que representan cada una de las pruebas, luego el SPI se encarga de enviar estos datos a la memoria flash. Una vez que la prueba se ha almacenado en su totalidad dentro de la memoria, esta comienza a enviar por una interfaz SPI instrucción por instrucción de la prueba del procesador. El CORE del procesador comienza a ejecutar internamente las instrucciones de la prueba y una vez que ha ejecutado todas las instrucciones, comienza a enviar los resultados por la interfaz UART formando un arreglo de bits.

Estos sets de instrucciones MIPS (aritméticas, lógicas, de desplazamiento, comparación, etc.), debido al diseño compacto del procesador, excitan casi en su totalidad los bloques internos del procesador. La figura 4.43 muestra un ejemplo de un set de instrucciones

aritmético, usado para comprobar el CORE. Todas estas pruebas están cargadas en archivos de texto con extensión TXT.

```

32  add t4,x0,x0
33  add t5,x0,x0
34  add t6,x0,x0
35  #-----
36  #-----suma 1000 con -1000 en s0 deberia quedar 0-----
37  #-----
38      addi ra,x0,1000
39      addi sp,x0,-1000
40      add s0,ra,sp
41  #-----
42  #-----suma hasta 16 potencias-----
43  #-----Prueba tambien usar registro entreda salida----
44  #-----Resultado en s1=16-----
45      addi s1,x0,2
46      add s1,s1,s1
47      add s1,s1,s1
48      add s1,s1,s1
49
50  #-----sumas normales-----
51  #-----s4=12000-----
52      addi s2,x0,400
53      addi s3,x0,100
54      add s4, s2, s3
55      addi s5,x0,900
56      add s6,x0,s5

```

Figura 4.43. Ejemplo de set de instrucciones aritméticas para el CORE

Pruebas Funcionales

La propuesta para el entorno que ejecute las pruebas, recopile los datos y luego los evalúe, se compone de dos VI's que se encarguen por separado de la escritura de las pruebas y de la lectura de la salida del TOP-LEVEL, luego los datos recopilados son enviados a TestStand donde serán comparados con arreglos de referencia que validan si la salida del CORE es correcta. La figura 4.44 muestra el diagrama de bloques de cómo se interconectarían las herramientas del entorno con el TOP-LEVEL del procesador.

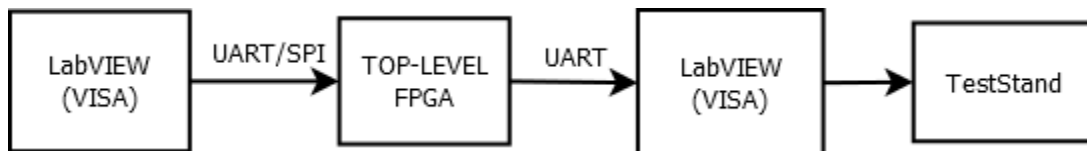


Figura 4.44. Diagrama de bloques del entorno de validación para el TOP-LEVEL

En el primer VI iniciaría cargando el archivo de texto donde están almacenadas las instrucciones MIPS, la función de LabVIEW que se encararía de esto sería “Read Delimited Spreadsheet” la cual lee líneas y columnas específicas de un archivo y los almacenaría en un

arreglo de caracteres. Antes de enviarse al TOP-LEVEL, el arreglo debe ser previamente convertido al formato ASCII para que le permita al canal UART recibir una mayor cantidad de bits en el menor tiempo posible, la velocidad de transmisión para el UART es de 9600 baudios. Luego por medio del módulo de comunicación VISA, el cuál es un módulo de LabVIEW dedicado a la comunicación serial, se enviarían los datos por UART a la memoria. La figura 4.45 muestra el diagrama de flujo de este primer VI.

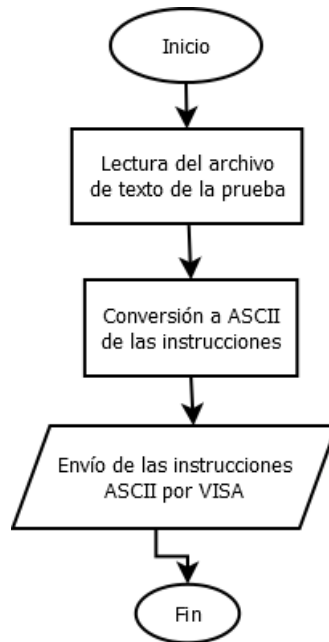


Figura 4.45. Diagrama de flujo del módulo de escritura para el procesador

Una ventaja que brinda el bloque de comunicación para pruebas diseñado por el DCI-Lab, es que las herramientas encargadas de la escritura y lectura de los sets de instrucciones no entran en contacto directo con el CORE del procesador, por lo cual no necesitan sincronizarse con el reloj del procesador y solo necesitan sincronizarse con la velocidad de transmisión de las interfaces UART.

Para la etapa de lectura de la salida del CORE, el segundo VI leerá por medio de VISA, las tramas de bits que envía el CORE, cada trama de bits es almacenada en un arreglo. En el momento en que la prueba termina, se envía una bandera que detiene el proceso de lectura y se cierra el arreglo, este arreglo es el que se envía a TestStand para su validación. La figura 4.46 muestra el diagrama de flujo para este VI.

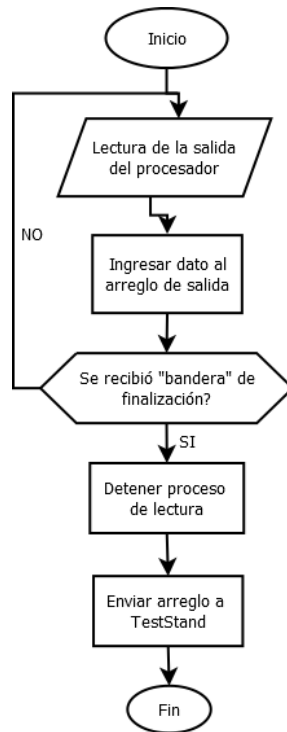


Figura 4.46. Diagrama de flujo para el módulo de lectura del procesador

Secuencia de pruebas para validación

La secuencia en TestStand estaría distribuida de manera que permita ejecutar ambos VI's de manera consecutiva, dependiendo de lo que el diseñador prefiera, se puede programar la herramienta para que cargue automáticamente los sets de instrucciones MIPS o que estos sean cargados de forma manual al inicio de cada secuencia.

Finalmente, para el módulo de lectura se utilizaría una prueba de tipo "Multiple Numeric Limit Test" para comprobar celda por celda el arreglo de salida del CORE. La regla de comparación utilizada sería la "EQ" (Equal), donde la herramienta validaría que cada celda del arreglo de salida del CORE sea igual a su celda correspondiente en el arreglo de referencia.

5 Conclusiones

Mediante el desarrollo de una metodología para validación post-silicio, fue posible la caracterización de circuitos integrados tipo combinacionales y secuenciales, que permitieron crear entornos de validación mínimos post-silicio que permitieron contrastar las características eléctricas y funcionales de los circuitos de caso de estudio, mediante una prueba de concepto.

A partir de la aplicación de un flujo de proceso de ingeniería, se logró desarrollar una metodología de diseño para entornos de validación post-silicio, que permite crear entornos “*custom*” o a la medida del circuito según el circuito de validación.

Mediante la aplicación de los conceptos de sistemas empotrados, combinando el hardware NI PXIe-1078 y sus módulos con los softwares LabVIEW v2017 y TestStand v2016, se logró ejecutar la metodología de diseño para crear módulos de medición, recopilación y comparación de las características eléctricas y funcionales de los circuitos.

Por medio de los módulos de medición, recopilación y comparación de datos de los entornos, se logró validar las características funcionales y eléctricas de circuitos combinacionales y secuenciales y se presentaron estos resultados en reportes estructurados generados por la herramienta TestStand.

A partir de las pruebas de validación desarrolladas, se comprueba que las herramientas que componen el entorno permiten analizar diferentes marcas de chips de una misma familia y los resultados de estas comprobaciones no se ven afectados por el cambio en la cantidad de iteraciones de cada prueba, cambio en la velocidad de reloj y casa matriz de circuitos integrados.

En relación a los resultados obtenidos según los requerimientos mínimos resumidos en la tabla 3.1, se concluye que:

- Con el entorno se logró validar circuitos cuya cantidad de pines se ajustaba a la cantidad requerida y en el caso del procesador, la conexión se puede realizar por medio de comunicación serial sin necesidad de pines.

- Se lograron hacer pruebas de características eléctricas principalmente tensiones, pero hubo problemas con las corrientes y los tiempos.
- Se lograron validar las características funcionales de ambos circuitos de las pruebas de concepto.
- Todas las pruebas, tanto físicas como eléctricas, fueron realizadas en una misma secuencia.
- El editor de secuencias permite añadir las pruebas deseadas y crear librerías de pruebas previamente creadas para su uso posterior.
- En la secuencia se realizan todas las pruebas de manera automática con diferentes cantidades de iteraciones.
- Por medio de la herramienta de TestStand, se genera un reporte en formato XML que resumen los resultados de la secuencia e indica si pasó o no las pruebas.

Mediante la exploración de las herramientas de software LabVIEW y TestStand y el hardware PXI, es posible llevar a cabo entornos de validación post-silicio para circuitos a medida y se ajusta de manera flexible el proceso de validación.

5.1 Recomendaciones

En función de aumentar la cantidad de tipos de chips que se pueden validar por medio de este entorno, a futuro se podrían crear bibliotecas de VI's de prueba para diferentes tipos de circuitos.

Para realizar pruebas a múltiples chips a la vez, es necesario realizar modificaciones al hardware del PXI, ya sea aumentar la cantidad de tarjetas DAQ o explorar las funcionalidades del módulo de "Switching" del PXI.

En caso de que se deseen realizar pruebas más específicas de tiempos y frecuencias, es necesario adquirir módulos de contadores para el PXI y actualizar la versión de software del LabVIEW para que pueda utilizar las funciones de tiempo más robustas.

Actualmente, no existe la posibilidad de combinar todo el entorno en un único VI, por lo que si se presentara un caso donde se necesitará dicha configuración, deberá buscarse otra solución.

Se debe diseñar y construir una placa de pruebas específica para pruebas de validación que permita reducir el margen de error y cumplir con las condiciones de la etapa de Power On Debug mencionada en el marco teórico de este documento.

6 Bibliografía

- [1] P. Mishra, R. Morad, A. Ziv y S. Ray, «Post-Silicon Validation in the SoC Era: A Tutorial Introduction,» IEEE Design and Test, vol. 34, nº 3, pp. 68-92, 2017.
- [2] I. Rangel, «Metodología de Monitoreo para Validación de Circuitos VLSI,» Instituto Politécnico Nacional, 2009. [En línea]. Available: <http://148.204.63.111/SABERv3/Repositorios/webVerArchivo/26122/1>.
- [3] P. Abramovici y P. Bradley, «A New Approach to In-System Silicon Validation and Debug,» EETimes, 2007. [En línea]. Available: https://www.eetimes.com/document.asp?doc_id=1276099&page_number=1.
- [4] D. Vázquez García de la Vega, «Diseño para testabilidad y tolerancia a fallos en circuitos analógicos,» Universidad de Sevilla, 1995. [En línea]. Available: <http://hdl.handle.net/11441/15705>.
- [5] E. Valderrama y C. Ferrer, «Diseño para la Testabilidad y BIST,» Universidad Autónoma de Barcelona, 2009. [En línea]. Available: <http://ocw.uab.cat/enginyeries/disseny-de-circuits-integrats-i/CAPITULO-10.pdf>.
- [6] «Device Under Test (DUT),» Techopedia, [En línea]. Available: <https://www.techopedia.com/definition/25924/device-under-test-dut>.
- [7] J. P. Tello Portillo, «Clasificación de Señales,» de Señales y Sistemas, Universidad del Norte, 2017, pp. 2-4.
- [8] D. O. Guaraglia y J. L. Pousa, «Analog Signals,» de Introduction to Modern Instrumentation - For Hydraulics and Environmental Sciences, De Gruyter, 2014, p. 64.
- [9] L. F. Monter Martínez y D. I. Rios Casañas, «Tipos de Señales,» de Comunicaciones en Redes, Universidad Autónoma del Estado de Hidalgo.

- [10] N. Cervantes y C. Pineda, «Fundamentos de Programación: Tipos de Datos,» Universidad Técnica del Norte, [En línea]. Available: http://www.utn.edu.ec/reduca/programacion/fundamentos/tipos_de_datos.html.
- [11] T. Noergaard, «What Is an Embedded System?,» de Embedded Systems Architecture - A Comprehensive Guide for Engineers and Programmers, Elsevier, 2005, pp. 5-7.
- [12] «Configuring PXI Systems Using MXI-Express,» National Instruments, [En línea]. Available: <http://www.ni.com/tutorial/4811/en/>.
- [13] «Qué es LabVIEW?,» National Instruments, [En línea]. Available: <http://www.ni.com/es-cr/shop/labview.html>.
- [14] «¿Qué es TestStand?,» National Instruments, [En línea]. Available: <http://www.ni.com/es-cr/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/what-is-teststand.html>.
- [15] «HD74LS08P,» Renesas, 2005. [En línea]. Available: https://www.renesas.com/us/en/doc/products/logic/001/rej03d0394_hd74ls08.pdf.
- [16] «Synertek 1981-1982 Data Catalog,» Bitsavers, [En línea]. Available: http://www.bitsavers.org/components/synertek/_dataBooks/Synertek_1981-1982_Data_Catalog.pdf.

7 Anexos

A. Mapeo de pines de los VI's

A continuación, se muestran las tablas de mapeo de pines para las pruebas del entorno de validación de la compuerta AND.

Tabla A.1. Distribución de pines de la prueba AND VCC

| 7408 | | SCB-68a | |
|--------|----------|---------|----------|
| Nombre | # de pin | Nombre | # de pin |
| VCC | 14 | AI0 | 68 |

Tabla A.2. Distribución de pines de la prueba AND OUTPUT VOLTAGE TEST

| 7408 | | SCB-68a | |
|--------|----------|---------|----------|
| Nombre | # de pin | Nombre | # de pin |
| 1Y | 3 | AI1 | 33 |
| 2Y | 6 | AI2 | 65 |
| 3Y | 8 | AI3 | 30 |
| 4Y | 11 | AI4 | 28 |

Tabla A.3. Distribución de pines de la prueba AND TRUTHTABLE TEST

| 7408 | | SCB-68 ^a | |
|--------|----------|---------------------|----------|
| Nombre | # de pin | Nombre | # de pin |
| 1A | 1 | P0.4 | 19 |
| 1B | 2 | P0.0 | 52 |
| 1Y | 3 | P0.5 | 51 |
| 2A | 4 | P0.1 | 17 |
| 2B | 5 | P0.6 | 16 |
| 2Y | 6 | P0.2 | 49 |
| 3A | 9 | P0.7 | 48 |
| 3B | 10 | P0.3 | 47 |

| | | | |
|----|----|------|----|
| 3Y | 8 | P2.3 | 46 |
| 4A | 12 | P2.2 | 45 |
| 4B | 13 | P1.0 | 11 |
| 4Y | 11 | P1.1 | 10 |

Las siguientes tablas corresponden al mapeo de pines para las pruebas del entorno de validación de la memoria SRAM.

Tabla A.4. Distribución de pines de la prueba SRAM VCC

| SRAM 2114 | | SCB-68a | |
|-----------|----------|---------|----------|
| Nombre | # de pin | Nombre | # de pin |
| VCC | 9 | AI0 | 68 |

Tabla A.5. Distribución de pines de la prueba SRAM OUTPOT VOLT TEST

| SRAM 2114 | | SCB-68a | |
|-----------|----------|---------|----------|
| Nombre | # de pin | Nombre | # de pin |
| I/O1 | 14 | AI1 | 33 |
| I/O2 | 13 | AI2 | 65 |
| I/O3 | 12 | AI3 | 30 |
| I/O4 | 11 | AI4 | 28 |

Tabla A.6. Distribución de pines de la prueba SRAM MEMORY TEST

| SRAM 2114 | | SCB-68a | |
|-----------|----------|---------|----------|
| Nombre | # de pin | Nombre | # de pin |
| CS | 8 | P0.4 | 19 |
| WE | 10 | P0.0 | 52 |
| A0 | 5 | P0.7 | 48 |
| A1 | 6 | P0.3 | 47 |
| A2 | 7 | P2.3 | 46 |
| A3 | 4 | P0.2 | 49 |

| | | | |
|------|----|------|----|
| A4 | 3 | P0.6 | 16 |
| A5 | 2 | P0.1 | 17 |
| A6 | 1 | P0.5 | 51 |
| A7 | 17 | P2.2 | 45 |
| A8 | 16 | P1.0 | 11 |
| A9 | 15 | P1.1 | 10 |
| I/O1 | 14 | P2.6 | 1 |
| I/O2 | 13 | P2.4 | 2 |
| I/O3 | 12 | P2.1 | 3 |
| I/O4 | 11 | P2.0 | 37 |

B. Diagramas de bloques de los VI's de pruebas

Los siguientes diagramas de bloques corresponden a los VI para activar y desactivar la fuente de alimentación del PXI.

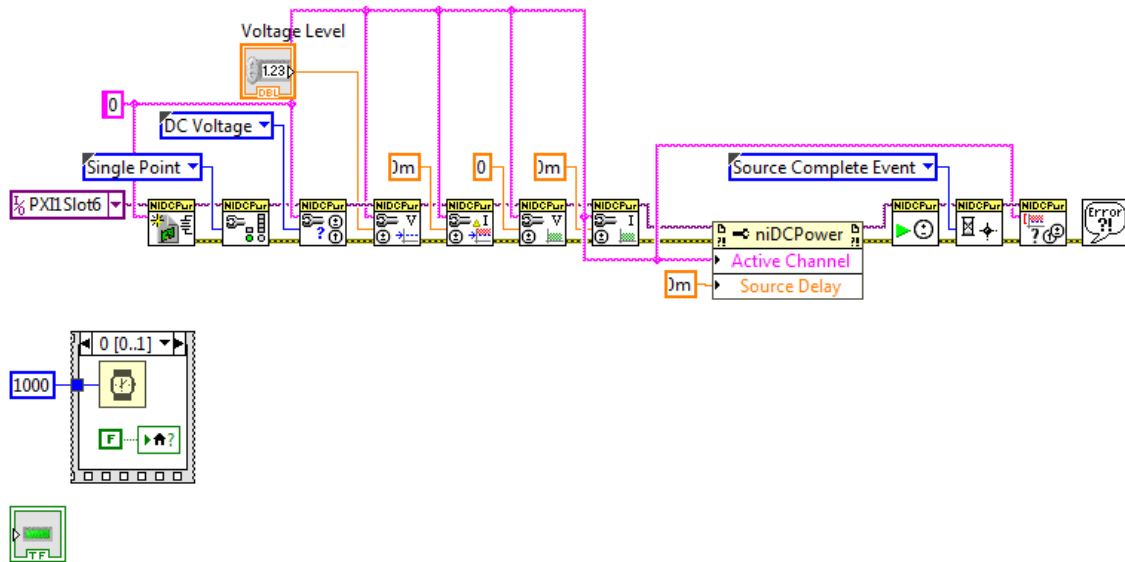


Figura B.1. Diagrama de bloques del VI para encender la fuente DC del PXI

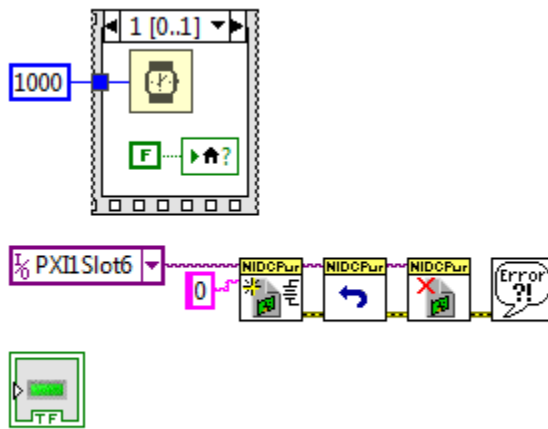


Figura B.2. Diagrama de bloques del VI para apagar la fuente DC del PXI

A continuación, se presentarán los diagramas de bloques de VI's que componen el entorno de validación para la compuerta AND.

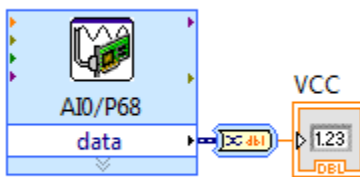


Figura B.3. Diagrama de bloques del VI AND VCC TEST

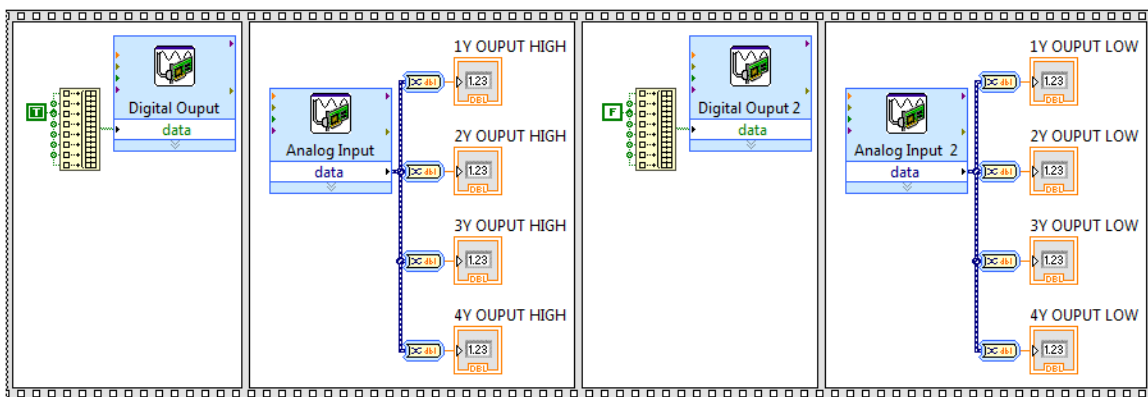


Figura B.4. Diagrama de bloques del VI AND OUTPUT VOLT TEST

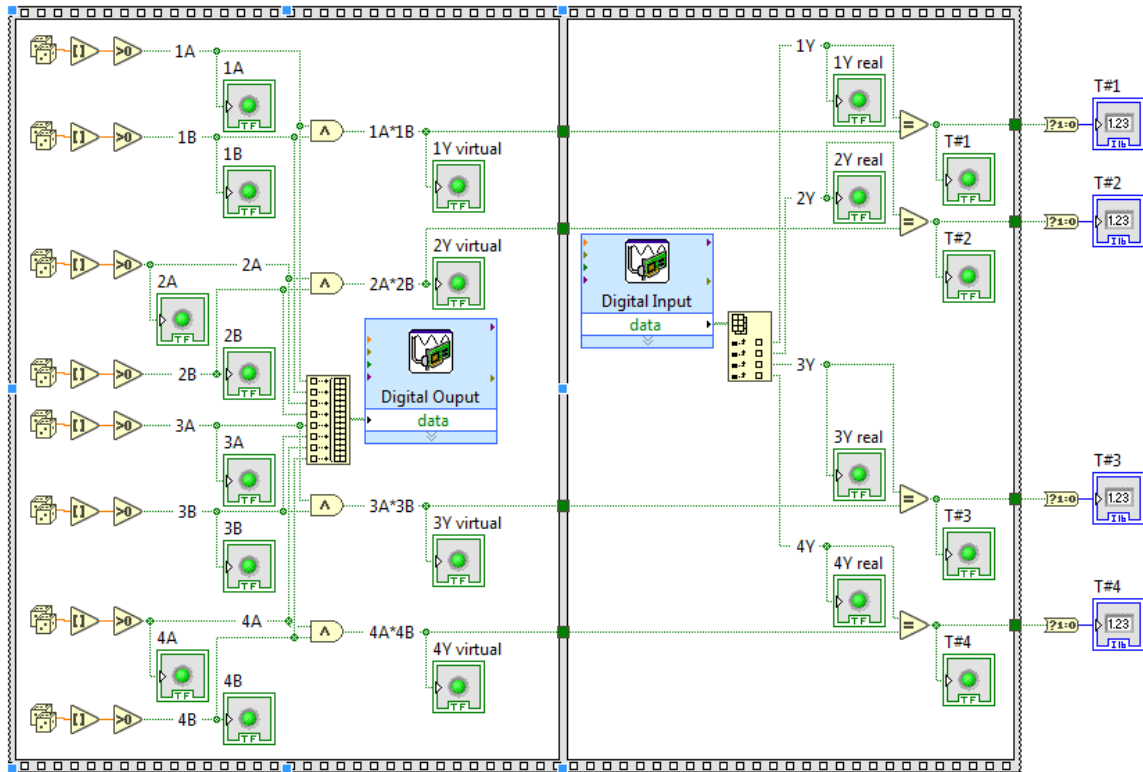


Figura B.5. Diagrama de bloques del VI AND TRUTH TABLE TEST

A continuación, se presentarán los diagramas de bloques de los VI's que componen el entorno de validación de la memoria SRAM.



Figura B.6. Diagrama de bloques del VI SRAM VCC TEST

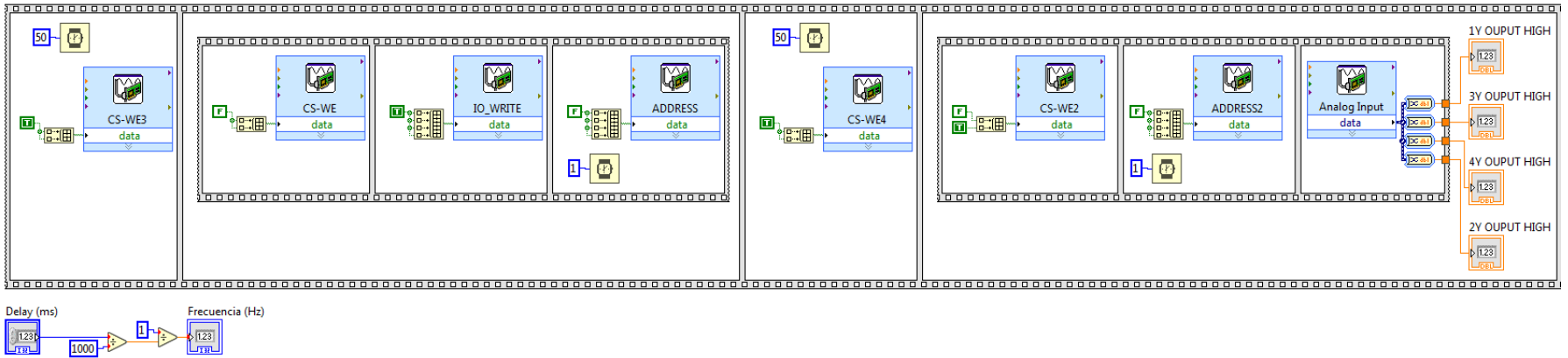


Figura B.7. Diagrama de bloques del VI SRAM OUTPUT VOLT TEST, parte 1

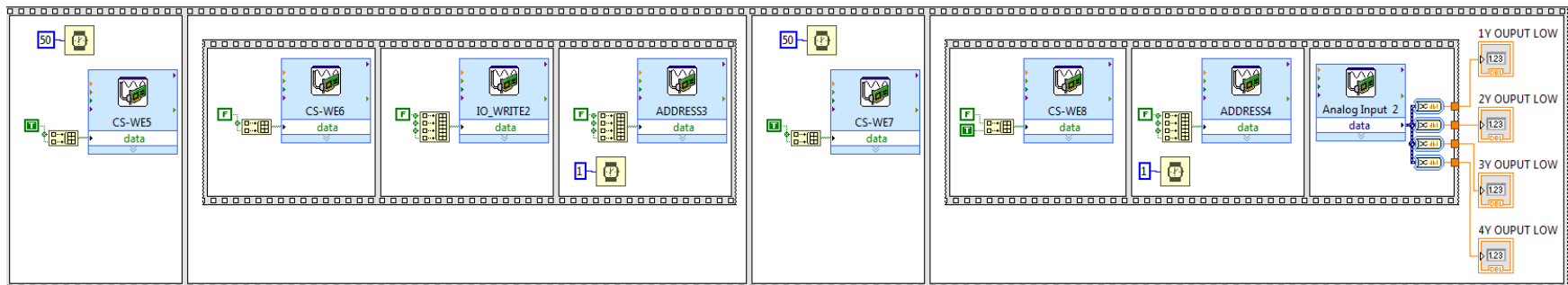


Figura B.8. Diagrama de bloques del VI SRAM OUTPUT VOLT TEST, parte 2

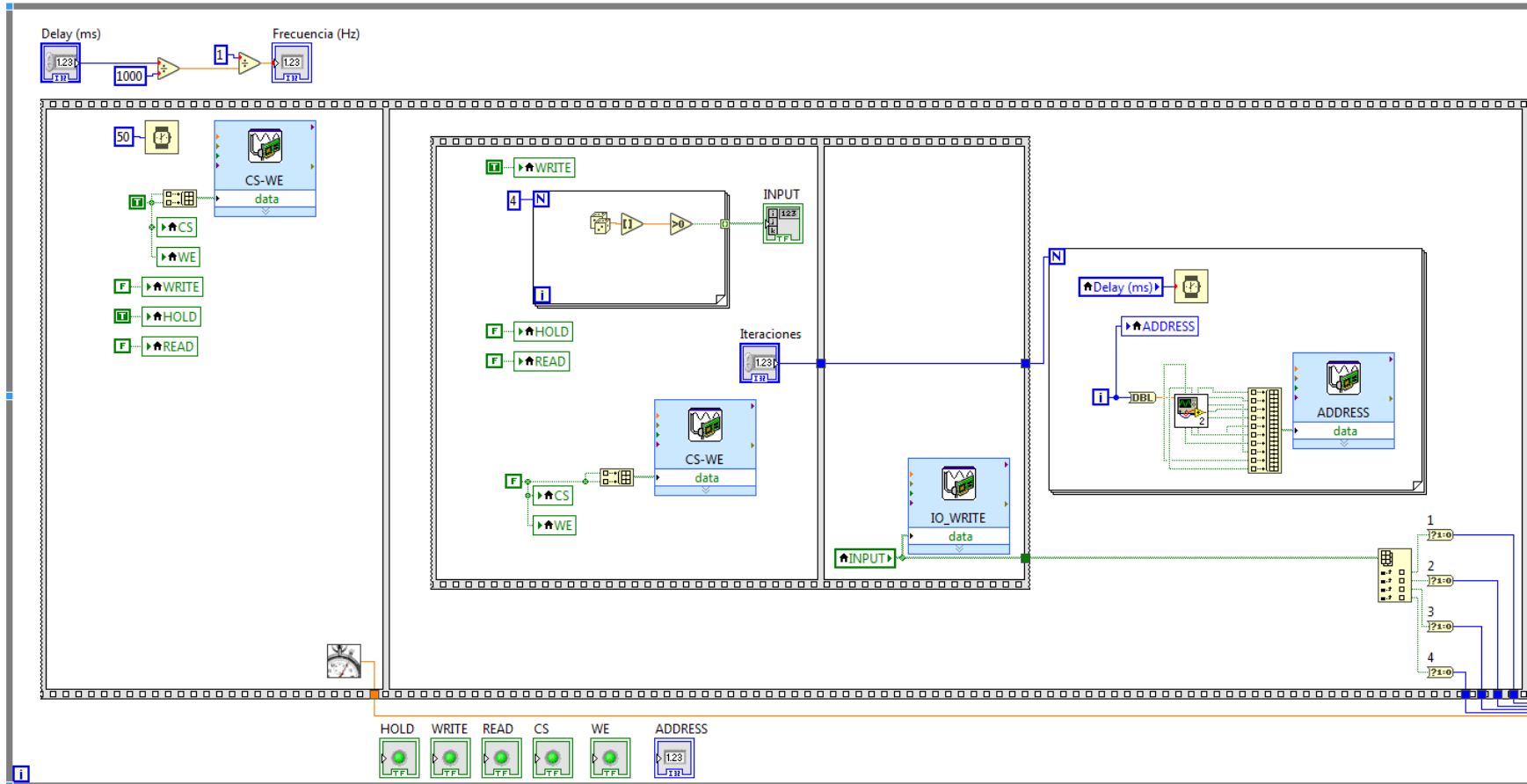


Figura B.9. Diagrama de bloques del VI SRAM MEMORY TEST, parte 1

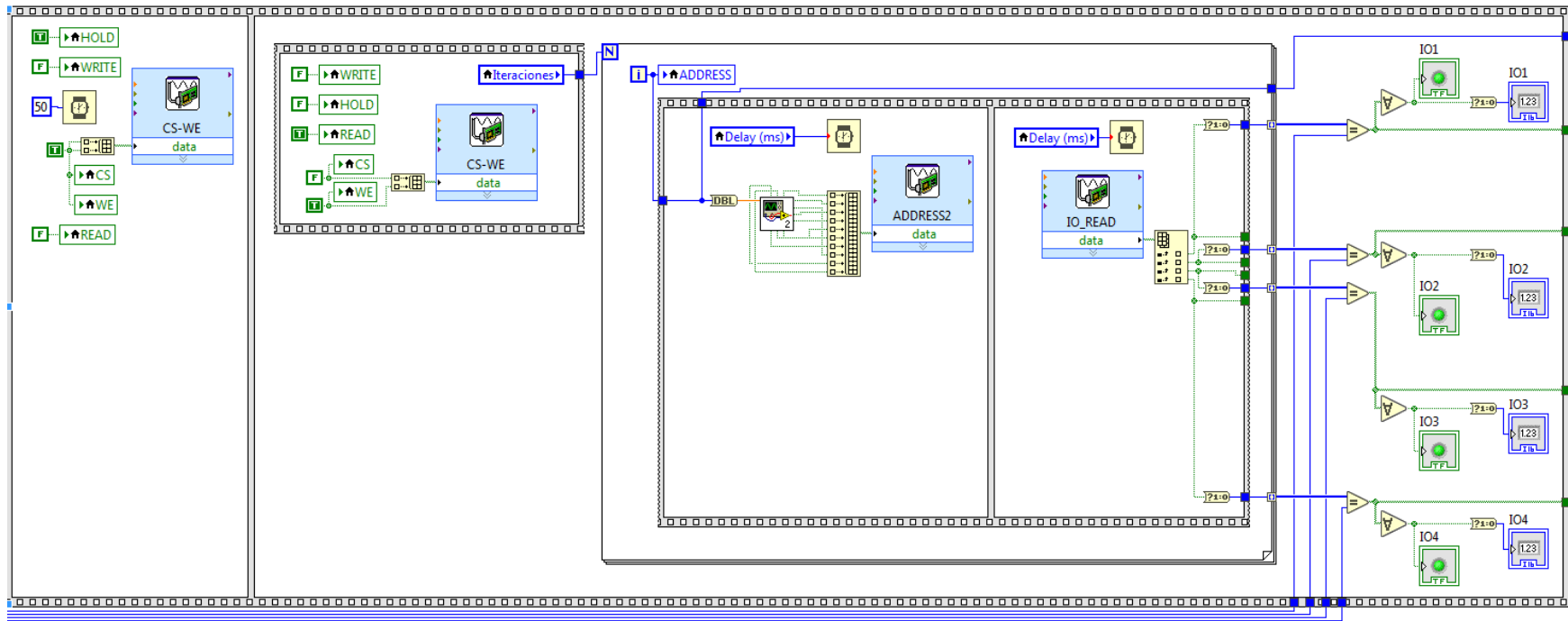


Figura B.10. Diagrama de bloques del VI SRAM MEMORY TEST, parte 2

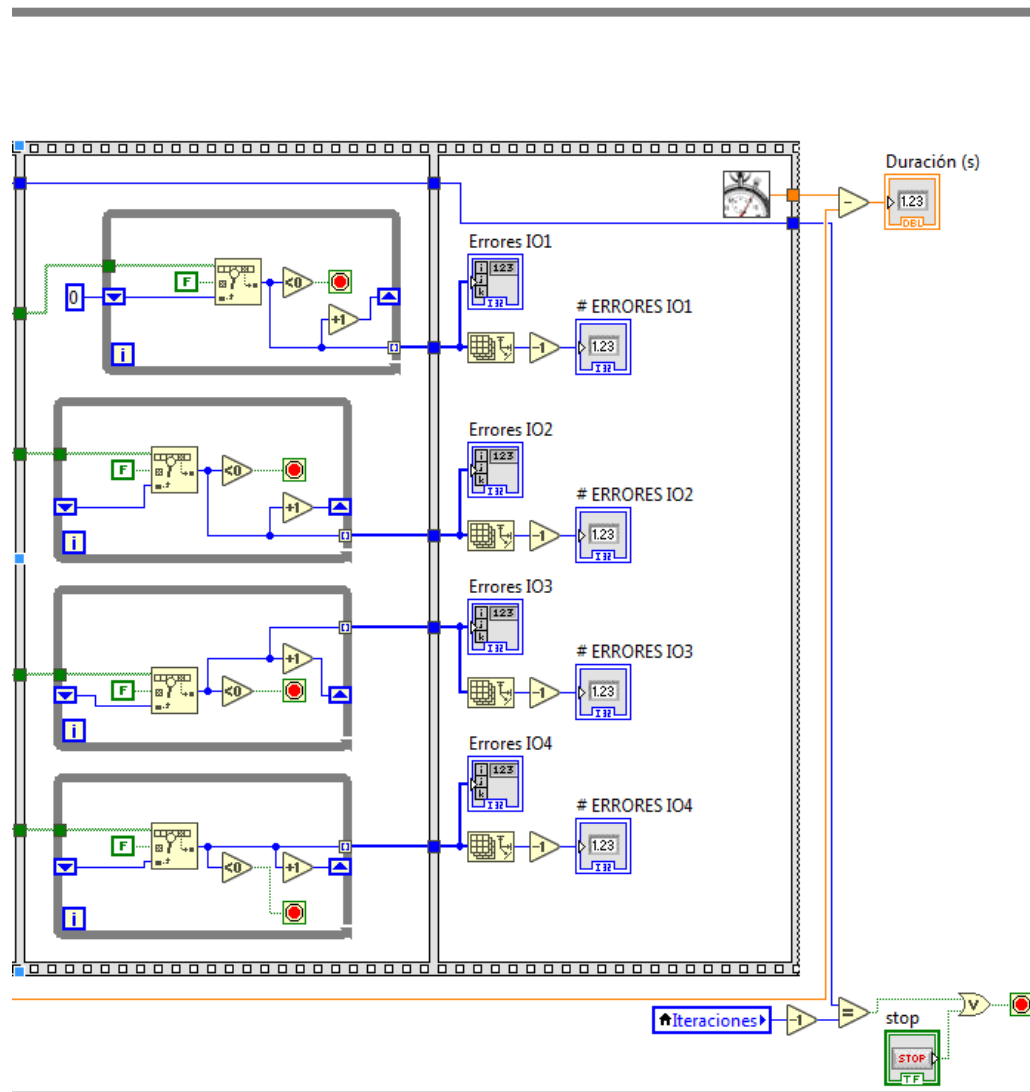


Figura B.11. Diagrama de bloques del VI SRAM MEMORY TEST, parte 2