

UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Protocolo Nanochip 2021**

Protocolo de trabajo de graduación presentado por Antonio Altuna  
Hernández, estudiante de Ingeniería Electrónica

Guatemala,

2021

## Resumen

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras vitae eleifend ipsum, ut mattis nunc. Pellentesque ac hendrerit lacus. Cras sollicitudin eget sem nec luctus. Vivamus aliquet lorem id elit venenatis pellentesque. Nam id orci iaculis, rutrum ipsum vel, porttitor magna. Etiam molestie vel elit sed suscipit. Proin dui risus, scelerisque porttitor cursus ac, tempor eget turpis. Aliquam ultricies congue ligula ac ornare. Duis id purus eu ex pharetra feugiat. Vivamus ac orci arcu. Nulla id diam quis erat rhoncus hendrerit. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Sed vulputate, metus vel efficitur fringilla, orci ex ultricies augue, sit amet rhoncus ex purus ut massa. Nam pharetra ipsum consequat est blandit, sed commodo nunc scelerisque. Maecenas ut suscipit libero. Sed vel euismod tellus.

Proin elit tellus, finibus et metus et, vestibulum ullamcorper est. Nulla viverra nisl id libero sodales, a porttitor est congue. Maecenas semper, felis ut rhoncus cursus, leo magna convallis ligula, at vehicula neque quam at ipsum. Integer commodo mattis eros sit amet tristique. Cras eu maximus arcu. Morbi condimentum dignissim enim non hendrerit. Sed molestie erat sit amet porttitor sagittis. Maecenas porttitor tincidunt erat, ac lacinia lacus sodales faucibus. Integer nec laoreet massa. Proin a arcu lorem. Donec at tincidunt arcu, et sodales neque. Morbi rhoncus, ligula porta lobortis faucibus, magna diam aliquet felis, nec ultrices metus turpis et libero. Integer efficitur erat dolor, quis iaculis metus dignissim eu.

## Antecedentes

La Universidad del Valle de Guatemala (UVG) a lo largo de los años se ha caracterizado por estar a la vanguardia en lo que es la investigación de nuevas tecnologías y metodologías en el área de las ciencias aplicadas. El Ing. Carlos Esquit, en el 2009, ingresó como nuevo director del departamento de Ing. Electrónica y Mecatrónica de la universidad y consigo trajo las destrezas y aprendizajes adquiridos en Estados Unidos en las ramas de micro y nano electrónica. Al iniciar, realizó una reforma al mapa curricular de Ing. electrónica para enseñar dichas disciplinas aprendidas en el primer mundo.

En el 2013, se inicia a impartir el curso de Introducción al diseño de sistemas VLSI. Al inicio las herramientas con las que se impartió este curso fueron gratuitas y permitían realizar diseños básicos a micro y nano escala de circuitos. Al año siguiente, el ingeniero formó una alianza con la empresa Synopsys. Esta se encuentra a la vanguardia en lo que se refiere a soluciones que facilitan el diseño en silicio de chips y su verificación. Esta proporcionó herramientas que actualmente utilizan los líderes de la industria de semiconductores. Con estas herramientas se realizó el primer diseño en silicio a nanoescala de 28nm, realizado por el Ing. Jonathan de los Santos en 2014.

En la nueva reforma curricular, que entró en vigencia en 2015, se añadieron los cursos de Nanoelectrónica 1 y 2. El fin de este cambio, que actualmente sigue vigente, es impartir teoría de VLSI y desarrollar en los alumnos competencias en el área de investigación del diseño de chips a micro y nano escala. Esto utilizando las herramientas que el software Synopsys le proporciona a la UVG. Con estas herramientas en el 2019 y 2020 se realizó el diseño para el primer chip con tecnología CMOS a nanoescala en Guatemala.

Los antecesores a este proyecto que trabajaron en el ciclo 2020 a 2021 en lo que se refiere a la síntesis física y verificación DRC (Design Rule Check) fueron el quipo de: Luis Abadilla y Matthias Sibrian. Los resultados obtenidos por ellos fueron diferentes simulaciones funcionales con lo que son las librerías de TSMC en los programas IC Validator y IC Compiler I. Adicionalmente dejaron documentación para utilizar los programas y scripts para replicar su trabajo.

## Justificación

La fabricación de Circuitos Integrados (CI) puede ser un proceso largo y riguroso. El elaborar un flujo de diseño para la fabricación de estos CI abre nuevas oportunidades para lo que es el diseño y manufactura de estos dispositivos facilitando y automatizando el proceso. En el mercado actualmente hay una gran variedad de CI de propósito general. Sin embargo el poder crear CI dedicados permite que los portadores de estos mejoren su desempeño ya que estarían centrados en cumplir la única funcionalidad. A diferencia de aquellos de propósito general que cuentan con múltiples herramientas, que si bien lo hacen más versátil, hacen que este no dedique el 100 % de sus recursos a una tarea.

Es por esto que el crear un flujo de diseño funcional, con herramientas de estado del arte, permite impulsar el desarrollo de la industria de semiconductores en Guatemala. Por medio tanto de la creación de plazas laborales que permitan el diseño y manufactura de estos dispositivos en el país así como la comercialización de estos productos.

Los beneficios inmediatos de este proyecto son el facilitar a quienes proceden el proceso del flujo de diseño. Para que se pueda invertir tiempo en únicamente a mantener actualizado el flujo de diseño y permitir que aquellos que se quieran beneficiar de este puedan crear proyectos de alto nivel en los departamentos de Electrónica, Macarrónica y Biomédica de la UVG. Y en un futuro que otras carreras puedan colaborar de forma interdisciplinaria para la creación de proyectos más ambiciosos y con un mayor alcance para beneficio de la comunidad y del país.

## Objetivos

### Objetivo General

Mejorar y concluir el proceso de diseño para la elaboración de un chip.

### Objetivos Específicos

- Nulla ut ex ut mauris pretium elementum.
- Suspendisse malesuada lectus nec nisi iaculis, in luctus turpis laoreet.
- In efficitur nisl vitae justo interdum, vitae condimentum lectus maximus.
- Morbi quis libero sit amet velit commodo tristique eu sed nisl.

## **Marco teórico**

### **0.1. VLSI**

Very Large Scale Integration o VLSI es el proceso completo por el cual se fabrica un circuito integrado (CI), los cuales están compuestos, actualmente, de millones de transistores a nanoescala. El proceso de diseño y simulación se le conoce como Design Flow. [1] Dicho proceso sigue una estructura top-down la cual implica que el proceso se lleva a cabo desde la jerarquía más alta (macro) hasta el proceso más específico (micros).

### **0.2. Flujo de Diseño [2]**

Es el proceso por el cual se obtiene el diseño de un CI fabricable y funcional. Este proceso puede dividirse en múltiples niveles de abstracción que conforman las diferentes capas del diseño de un CI. Las ocho capas en las que se divide esta investigación son: sistema, algoritmo, arquitectura, lógica, física y singoff. A continuación se describe a mayor detalle cada uno de ellos. Cabe tomar en cuenta que la necesidad de dividir todo el proceso de fabricación en capas surge ya que la complejidad de diseño es tal que requiere de especialización en diferentes disciplinas. El segmentar en capas permite que cada especialista trabaje en un nivel de abstracción específico aportado al flujo de diseño flexibilidad, cooperación y calidad.

#### **0.2.1. Diseño en la capa Sistema**

Esta es la jerarquía más alta de diseño, el objetivo que se desea conseguir en esta etapa es diseñar un sistema que cumpla con las necesidades básicas de su aplicación. Las principales características en las que se especializa esta capa son: funcionalidad, desempeño, condiciones de trabajo, dimensiones físicas, empaquetado, pinout, tecnología de fabricación, costos, la potencia consumida y los protocolos de comunicación interna y externa.

#### **0.2.2. Diseño en la capa Algoritmo**

Esta capa busca definir una estructura en software que permita una implementación a hardware funcional y sencilla. En esta etapa es importante considerar la capacidad de, computo del CI, la cantidad de memoria que este requiere, se define la arquitectura del set de instrucciones (ISA) por sus siglas en inglés [3]. En este punto debe evaluarse que tan complejo, que tantos recursos computacionales y que tanta exactitud se requiere según los objetivos establecidos en la capa superior como los protocolos de comunicación y las tareas que debe realizar.

### 0.2.3. Diseño en la capa Arquitectura

Así como la capa superior busca distribuir los recursos computacionales de forma eficiente, esta capa busca distribuir de forma eficiente los recursos físicos (hardware) para cada una de las tareas que se van a realizar. Aquí se debe de prestar atención a los objetivos del diseño en la capa de sistema como: desempeño, costo, potencia, dimensiones, etc... En este punto la operación de la máquina (MO), por sus siglas en inglés, toma lugar que se refiere a como el hardware implementa el ISA definido en la capa superior [3]. La definición de datapaths, tamaño de los buses, memorias de acceso random (RAM) y su relación con los demás componentes son las tareas de prioridad en esta capa.

### 0.2.4. Diseño en la capa Lógica

En esta capa se busca generar una descripción lógica y hacer una síntesis de la misma para generar lo que se llama un **netlist**. Ambas tareas críticas para el diseño del CI.

Ya con una arquitectura que define todo el proceso de interacción del hardware se requiere definir la conectividad de cada módulo en el nivel de abstracción más alto de esta capa que se traduce en *black boxes* las cuales contienen entradas, salidas y una sincronización temporal definida. La cual se puede describir utilizando un lenguaje descriptor de hardware o HDL por sus siglas en inglés. Los lenguajes más comunes para poder describir el hardware son: Verilog y VHDL. Estos módulos deben ser simulados y verificados para asegurar su funcionalidad.

El siguiente paso es la traducción de estos bloques de HDL a elementos de bajo nivel de un circuito y como estos están interconectados entre sí. A el resultado de esta traducción se le conoce como **netlist**. Herramientas de síntesis permiten automatizar estos procesos lo que facilita esta tarea de "traducir" de HDL a una netlist.

Los netlists utilizan algo llamado *cell librarys* que contienen los elementos más elementales de un circuito. Los transistores son definidos y conforman subcircuitos que representan compuertas lógicas las cuales en conjunto con los elementos básicos como capacitores son descritos tanto sus propiedades como su interconexión. Esto se puede tanto simular como validar utilizando una herramienta de simulación **spice**.

### 0.2.5. Diseño en la capa Física

A partir de este punto cada subcircuito descrito en el netlist es colocado en un espacio físico en una oblea de material semiconductor (Silicio) la cual está conformada de las diferentes difusiones, metales, pines así como sus interconexiones. Debido a que el fabricante tiene requerimientos de diseño, que se deben cumplir para garantizar la funcionalidad del diseño es imperativo que se tomen en cuenta y se verifiquen de forma rigurosa. Ya que este proceso toma en cuenta tantos factores como: rendimiento, área, confiabilidad, potencia entre otros... el proceso se divide en diferentes secciones: Partitioning , FloorPlanning, Power and Ground Routing, Placement, Clock Network Synthesis, Global Routing, Detailed Routing y Timing Closure de las cuales se discutirán a mayor detalle. Al final de todo este proceso un **netlist** asociado es generado el cual contiene todos los cambios y restricciones colocadas en

los diferentes pasos de esta capa de síntesis física.

### **0.2.6. Diseño en la capa Signoff y Verificación Física**

Una vez se culmina el diseño este debe de ser verificado de forma rigurosa para asegurar que este cumpla con los estándares de fabricación, que concuerde el diseño en silicio con el de el circuito propuesto y cualquier problema debe ser solucionado en la capa anterior de diseño. Las verificaciones que se realizan se mencionan a continuación:

#### **Design Rule Checking (DRC)**

Verifica que el layout cumpla con todos los requerimientos de fabricación impuestos por el fabricante directamente relacionadas a la tecnología que se está trabajando.

#### **Layout vs. Schematic (LVS)**

Verifica la funcionalidad del diseño, el **netlist** generado por la capa superior, síntesis física, y el **netlist** generado por la capa de síntesis lógica son comparados para verificar si ambos concuerdan en términos de funcionalidad.

#### **Parasitic Extraction**

Este deriva de los componentes geométricos, a nivel nanométrico, sus parámetros eléctricos (Capacitancias y Resistencias) para verificar las características eléctricas del circuito.

#### **Antenna Rule Checking**

Esta fase sirve para prevenir los efectos de antena en el circuito, ya que este al tener electrones movilizándose por un conductor puede comportarse como una antena. Esto representa un problema ya que puede causar daños a los transistores por acumulaciones de carga en metales que no necesariamente esten conectados a una unión PN.

#### **Electrical Rule Checking (ERC)**

Verifica la calidad de la conexión de tierra y alimentación así como los tiempos de transmisión de señales. Adicionalmente verifica cargas capacitivas y correcta conexión de los **fanouts**

### 0.2.7. Diseño en la capa de Fabricación [3]

Una vez el diseño ha pasado por todas las verificaciones de diseño y el resultado es CLEAN o PASS en todos, que regularmete es un archivo en formato **GDS** entonces se envía a una fábrica que se dedica a la fabricación de chips en silicio. En el caso del proyecto es a TSMC. El proceso en donde se manda el archivo a la fábrica se le conoce como **tapeout**. Aquí se generan las máscaras y por el proceso de fotolitografía patrones expuestos a los rayos láser se graban en el silicio, así como los diferentes dopajes. El diseño se hace sobre obleas de silicio redondas las cuales varían en tamaño desde los 200mm hasta los 300mm. Por último se clasifican según su estado funcional o defectuoso, establecido por las pruebas de potencia y velocidad que se establecieron.

### 0.2.8. Diseño en la capa de Empaquetado y Pruebas

Aquí se cortan los CI individualmente de la oblea de silicio donde se fabricaron, a los que cada CI se le conoce como **die** y este es empaquetado según la necesidad del cliente. Los empaquetados pueden ser: DIP, PGS, BGA, etc.. Se conectan los pines dentro del empaquetado con el **die** ya sea con bonding wires o solder bumps y se sella. Y como punto final del flujo de diseño, este se prueba en un laboratorio por el cliente para ver que este funcionando de la forma deseada y con los requerimientos propuestos. Concluyendo el flujo de diseño del CI.

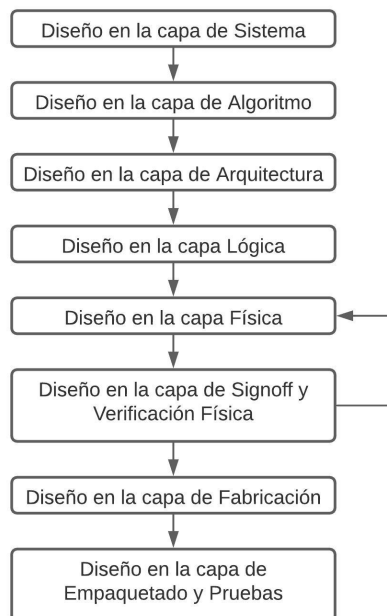


Figura 1: Proceso de Design Flow completo

### 0.3. Síntesis Física y Design Rule Check (DRC)

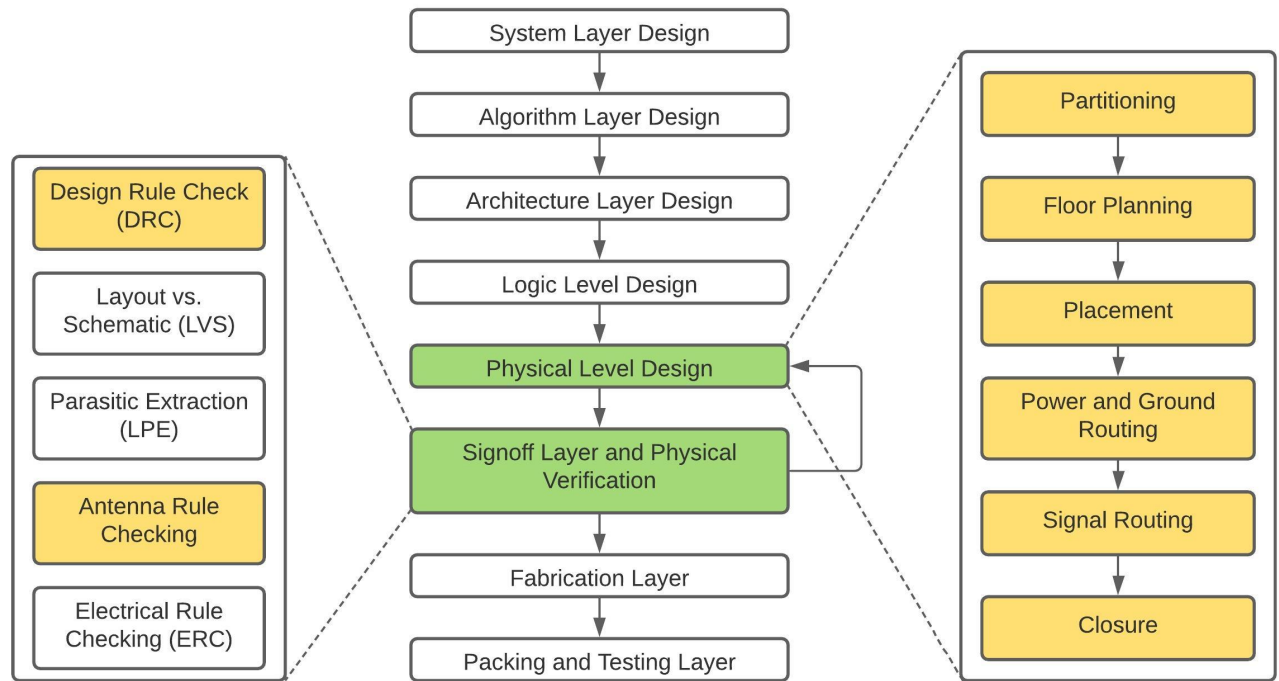


Figura 2: Proceso de Design Flow con las partes a realizar

### 0.4. Synopsys

Synopsys es una herramienta de diseño y simulación para cada una de las diferentes etapas del design flow. En el caso las herramientas proporcionadas por el departamento de Ing. Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala, de las cuales se aprenden a utilizar en los cursos de Nano Electrónica 1 y 2 son: HSpice, Wave Viewer, IC Compiler, IC Compiler II, Custom Compiler, Verdi3, VCS, Design Vision, Nanotime, IC Validator, StarRC, Formality, Milky Way entre otros. En las siguientes subsecciones se desarrolla acerca de las herramientas que son de interés para el desarrollo del proyecto de graduación. [4]



#### 0.4.1. IC Validator

#### 0.4.2. IC Compiler I

#### 0.4.3. IC Compiler II

### Metodología

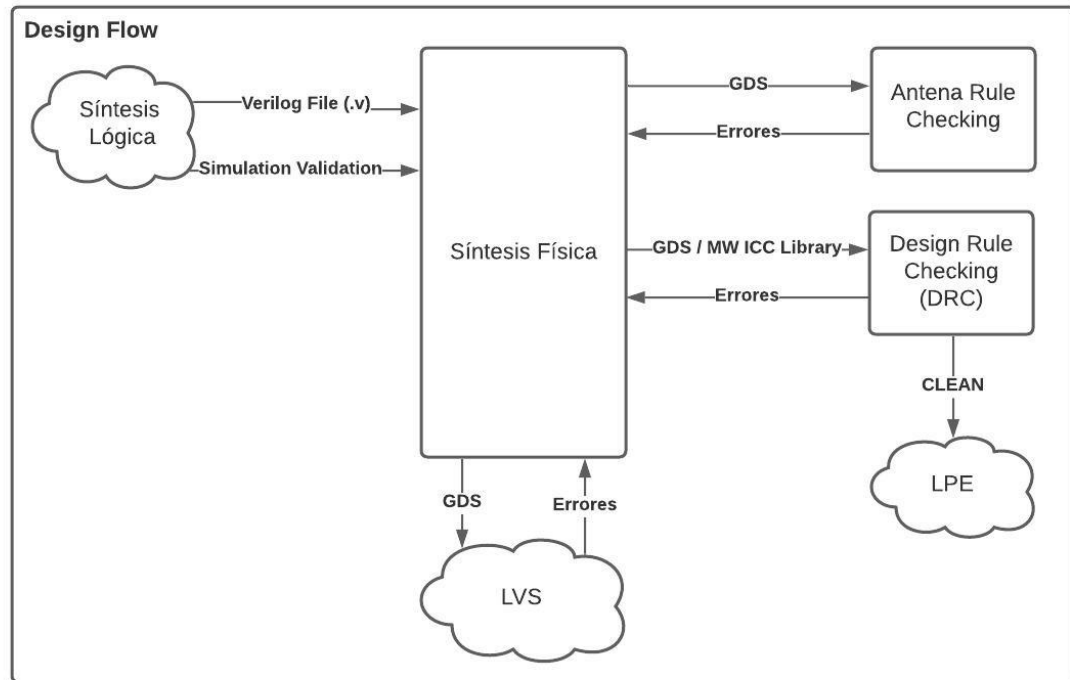


Figura 3: Input Output design flow diagram

### Cronograma de actividades

### Índice preliminar

### Referencias

- [1] L. Nájera, "Implementación de circuitos sintetizados a nivel netlist a partir de un diseño en lenguaje descriptivo de hardware como primer paso en el flujo de diseño de un circuito integrado.," en *Trabajo de Graducacion, Modalidad Tesis*, Facultad de Ingeniería Universidad del Valle de Guatemala, 2019.

- [2] M. Sibrian, “Verificación de reglas de diseño (DRC) para el desarrollo de un flujo funcional de un circuito integrado con tecnología nanométrica,” en *Trabajo de Graducacion, Modalidad Tesis*, Facultad de Ingeniería Universidad del Valle de Guatemala, 2020.
- [3] C. Esquit, *Computer Architecture Lecture 1: Introduction and Five Components of a Computer*, jul. de 2020.
- [4] J. de los Santos, “Diseño de un sumador/restador completo de 32 bits con tecnología CMOS en un proceso de 28 nanómetros usando aplicaciones de diseño de la empresa Synopsys.,” en *Trabajo de Graducacion, Modalidad Tesis*, Facultad de Ingeniería Universidad del Valle de Guatemala, 2019.