

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



## 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, Verificación ERC (Electrical Rule Check), Antenna Rule Check y verificación DRC (Design Rule Check) fueron el quipo de: Luis Abadilla, Marvin Flores y Matthias Sibrian. Los resultados obtenidos por ellos fueron diferentes simulaciones funcionales, con 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 la utilización del flujo de diseño a quienes preceden esta investigación. 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. Además, a largo plazo, 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 UVG 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

### 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 Flujo de Diseño. [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 (micro).

#### 0.1. 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 seis capas en las que se divide este proceso son: sistema, algoritmo, arquitectura, lógica, física y *singoff*. A continuación se describe a mayor detalle cada uno

de ellas. Cabe resaltar 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.

#### 0.1.1. 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 la necesidad de comunicación interna y externa.

#### 0.1.2. 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 conjunto 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.

#### 0.1.3. 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 y 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.

#### 0.1.4. Diseño en la capa Lógica

En esta capa se busca generar una descripción lógica del circuito 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 se traduce en *black boxes* las cuales contienen entradas, salidas y una sincronización temporal definida (reloj). 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 lógica permiten automatizar estos procesos, lo que facilita esta tarea de traducir de **HDL** a un *netlist*.

Los *netlists* utilizan algo llamado *cell libraries* 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, resistencias, entre otros... son descritos tanto sus propiedades como su interconexión. Esto se puede simular y validar utilizando una herramienta de simulación *spice*.

### 0.1.5. 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 y 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 factores como: rendimiento, área, confiabilidad, potencia entre otros... el proceso se divide en diferentes secciones: *Partitioning*, *FloorPlanning*, *Placement*, *Power and Ground Routing*, *Signal Routing* y *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.

### 0.1.6. 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, etc... 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 (LPE)***

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 estos pueden acumularse generando descargas no deseadas. Esto representa un problema ya que puede causar daños a los transistores o reducir su tiempo de vida.

### ***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 la correcta conexión de los *fanouts*

## **0.1.7. Diseño en la capa de Fabricación [3]**

Una vez el diseño ha pasado por todas las verificaciones y el resultado es **CLEAN** o **PASS** en todas, que es un archivo en formato **.GDS2** entonces se envía a una fábrica que se dedica a manufacturar chips en silicio. En el caso del proyecto, el fabricante es **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.

## **0.1.8. Diseño en la capa de Empaquetado y Pruebas**

Aquí se cortan los **CI** individualmente de la oblea de silicio donde se fabricaron. A 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. 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**

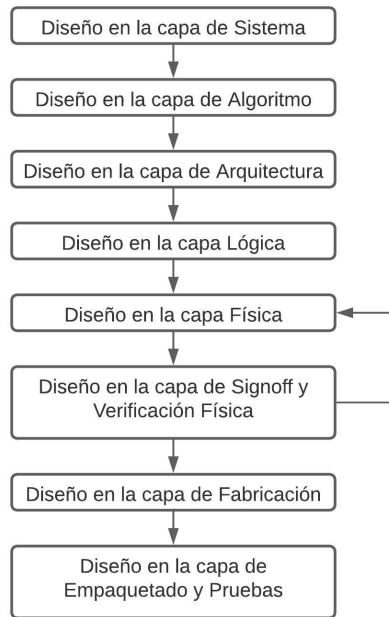


Figura 1: Proceso de flujo de diseño completo

## 0.2. Síntesis Física, *Design Rule Check (DRC)*, *Electrical Rule Check (ERC)* y *Antena Rule Check*

### 0.2.1. Síntesis Física

La síntesis física es una de las partes fundamentales para el desarrollo de este proyecto, ya que aquí se transformará la síntesis lógica a un conjunto de componentes físicos posicionados en silicio y unidos a través de *interconnects*. Esta etapa de diseño está segmentada en diferentes secciones las cuales se describen a continuación:

#### *Partitioning*

En esta etapa se secciona el circuito en subcircuitos con el fin de que estos puedan luego ser posicionados y colocados de forma estratégica.

#### *Floor Planning*

En esta etapa se hace una relación espacial entre los pines de entradas y salidas del **CI** y los subcircuitos. Esto se hace con la finalidad de establecer prioridades a las secciones

del circuito original que deben estar más cerca de estos puertos ya que se desea minimizar retardos y material utilizado.

### *Placement*

En esta etapa se plantean secciones del espacio que se va a utilizar para realizar el **CI**. En el caso de este proyecto se cuentan con 2mm de espacio para la fabricación del circuito deseado. Las partes seccionadas en la primera etapa se les asigna de forma estratégica una posición física en el silicio con el fin de obtener los mejores resultados en cuanto a la minimización de retardos y distancia entre las entradas y salidas de estos subcircuitos con los demás.

### **Power and Ground Routing**

En esta etapa se hace un ruteo de los planos de alimentación y tierra. Esto se hace de primero ya que todo circuito lógico compuesto de **MOSFETs** tendrá la necesidad de estos, lo que hace crítico que estas señales se hagan primero.

### **Signal Routing**

Se generan *interconnects* entre todos los circuitos lógicos. El software utilizado permite optimizar estas trayectorias a manera de que utilicen menor material. Ya que entre más distancias y más material la cantidad de parásitos incrementará lo que impacta directamente con el rendimiento del **CI**.

### *Closure*

Este es el paso antes de pasar a las verificaciones físicas, en esta fase se realizan las últimas revisiones del *layout*. Se puede realizar un **DRC** básico que proporciona la herramienta de IC Compiler II.

#### **0.2.2. DRC [2]**

*Design Rule Check*, esta etapa es crítica y juega un papel fundamental en conjunto con la síntesis física. Aquí se comparan los resultados obtenidos de la síntesis física con un *runset*, que en nuestro caso es proporcionado por **TSMC**, el cual contiene todas las restricciones de fabricación. Aquí se hace una exhaustiva revisión de las normas como el espaciado de metales, las interconexiones, el tamaño de los *pads*, etc... se cumplan y concuerden con todas las normas de **TSMC**. Esta etapa no trata acerca de la funcionalidad ni la coherencia entre el diseño lógico y físico que es tarea del **LVS**. Cualquier problema resultante de esta verificación debe ser gestionado por la capa de diseño superior (síntesis física) y de forma iterativa solucionar cualquier problema que se presente hasta obtener un resultado **CLEAN**.



### 0.2.3. ERC [4]

*Electrical Rule Check* es una etapa de la capa se *singoff* la cual permite verificar que el rendimiento eléctrico del circuito cumpla con las especificaciones del fabricante. Las reglas que verifica esta etapa son las siguientes:

#### *Soft Connect Check*

Esta regla establece si la conectividad (unidireccional) de las difusiones que se encuentran en el *N-Well* y la capa superior es la especificada.

#### *Path Check*

Esta regla valida lo siguiente: Nodos con un *path* conectado a alimentación pero no tierra, Nodos con un *path* a tierra pero no a alimentación, Nodos sin *path* a tierra o alimentación o Nodos sin un *path* a cualquier otra *net*.

#### *PTAP/NTAP Connectivity Check*

Verifica si el *PTAP* se conecta a alimentación y el *NTAP* a tierra.

#### *MOSFET Poer and Ground Check*

Verifica que para cualquier **MOSFET**, ya sea tipo N o P, *source* **MOSFET** y *drain* **MOSFET** estén conectados correspondientemente a alimentación y a tierra.

#### *Gate Directly Connected to Power or Ground*

Esta regla sirve para verificar que exista protección contra descargas electro-estáticas. Si el *gate* de un **MOSFET** tipo P se conecta a alimentación de forma directa podría dañarse el *gate*, así como un **MOSFET** tipo N conectado a tierra directamente.

#### *Floating Gate*

Verifica si los contactos están interfiriendo con el *poly gate*

#### *Floating Well*

Verifica si existe alguna conexión entre el *N-Well* o *P-Subrate* y alimentación o tierra.

#### 0.2.4. *Antenna Rule Checking* [4]

Esta verificación busca evitar que el **CI** tenga efectos de antena que pueden dañar las terminales de los transistores (*gate*, *drain*, *source* o *bulk*) durante el proceso de fabricación, que puede ocurrir en s el grabado de los caminos por rayos plasma. Esto ya que se pueden acumular cargas en los cables metálicos no deseadas y que estén conectadas a nodos de uniones PN. Algunos de los efectos presentes pueden ser: daño de oxido inducido por plasma (efecto antena), e inducción de corriente por altas temperaturas. Efectos que pueden quemar o reducir el tiempo de vida de un transistor afectando el rendimiento y tiempo de vida del **IC** completo.

Esta verificación al igual que el **DRC** cuenta con un *runset* específico y utiliza el software de IC Compiler II. Las reglas de antena se centran en especificar el área máxima de metal que se puede conectar a un gate sin el source o drain para actuar como elemento de descarga. Las relaciones entre gate y los demás metales son: 100:1 hasta 5000:1 para que la carga sobre el metal no dañe la compuerta. Algunas formas de solucionar estos problemas pueden ser: seccionar el metal en conexiones de metales más pequeños para reducir la potencia del rayo plasma durante la fabricación o la colocación de un diodo entre el metal y tierra para evitar la conducción de corriente y que esta pueda causar una descarga no deseada por las altas temperaturas.

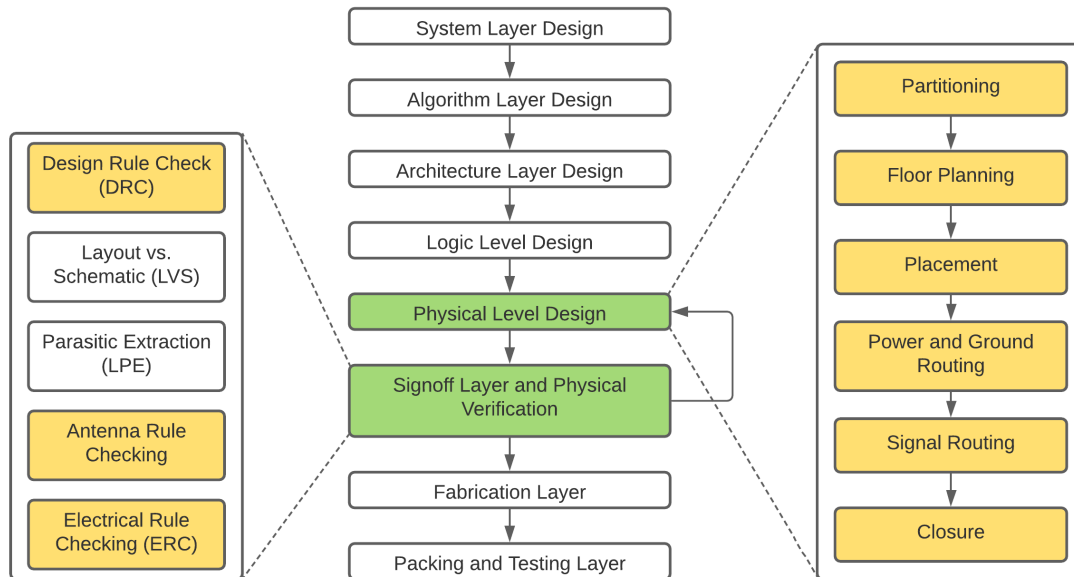


Figura 2: Proceso de flujo de diseño con las partes a realizar

### 0.3. *Electronic Design Automation (EDA)*

La industria **EDA** existe con el fin de proporcionar herramientas de software para diseñadores electrónicos, para automatizar los procesos de diseño y fabricación de **CI**s. Esto permite que los precios para la fabricación de estos dispositivos disminuya, además de la especialización en cada una de las herramientas de software y separación del proceso de fabricación con el de diseño. Esto genera equipos de trabajo especializados en el área de diseño de **CI**s y por otro lado las compañías que fabrican dichos dispositivos pueden proporcionar las restricciones y normas de diseño con las que pueden fabricar los dispositivos. Si bien ambas disciplinas se separan estas convergen con las herramientas **EDA**, siendo el puente entre los diseñadores y la industria de semiconductores, reduciendo costos, generando equipos especializados y creando una nueva industria en el mundo.[5]

Actualmente las dos compañías más grandes en la industria **EDA** son Cadence y Synopsys. Para fines de esta investigación se estarán utilizando las herramientas **EDA** de Synopsys. Esta cuenta con todas las herramientas para completar el flujo de diseño propuesto en la Figura 1 hasta la etapa de *Signoff and Physical Verification*. Esto se debe a que Synopsys no es una empresa que se dedica a la fabricación de **CI**s. Además como se mencionó anteriormente este es el puente entre el fabricante y el diseñador, por lo que las restricciones de diseño y fabricación han sido proporcionadas por **TSMC** la empresa a la que se mandará a hacer el **CI**, en donde se podrán concluir las últimas dos etapas del flujo de diseño y obtener el **CI** físico.[6]

### 0.4. Synopsys

Synopsys es una herramienta de diseño y simulación para cada una de las diferentes etapas del flujo de diseño. Estas herramientas aon proporcionadas por el departamento de Ing. Electrónica, Mecatrónica y Biomédica de la UVG. 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 otras. En los siguientes apartados se desarrolla acerca de las herramientas que son de interés para el desarrollo del proyecto de graduación en lo que serán las etapas de Síntesis Física y en la etapa de *Signoff* y la Verificación Física: **DRC**, **ERC** y *Antenna Rule Checking* [7]

#### 0.4.1. IC Compiler II [8]

La sucesora a la herramienta IC Compiler I, es una de las herramientas líder para lo que es Place and Route. Esta herramienta es esencial para la síntesis física. En esta herramienta permite procesar el *verilog* proveniente de la síntesis lógica para convertirlo en un *layout* que represente el circuito físico en silicio, el cual será luego verificado.

### 0.4.2. IC Validator [9]

IC Validator es una de las herramientas de Synopsys para la etapa de *Signoff* y Verificación física. Este software permite realizar **DRC**, *Antenna* y **ERC**, verificaciones necesarias para asegurar que el *layout* generado en la etapa anterior pueda ser fabricado.

## Metodología

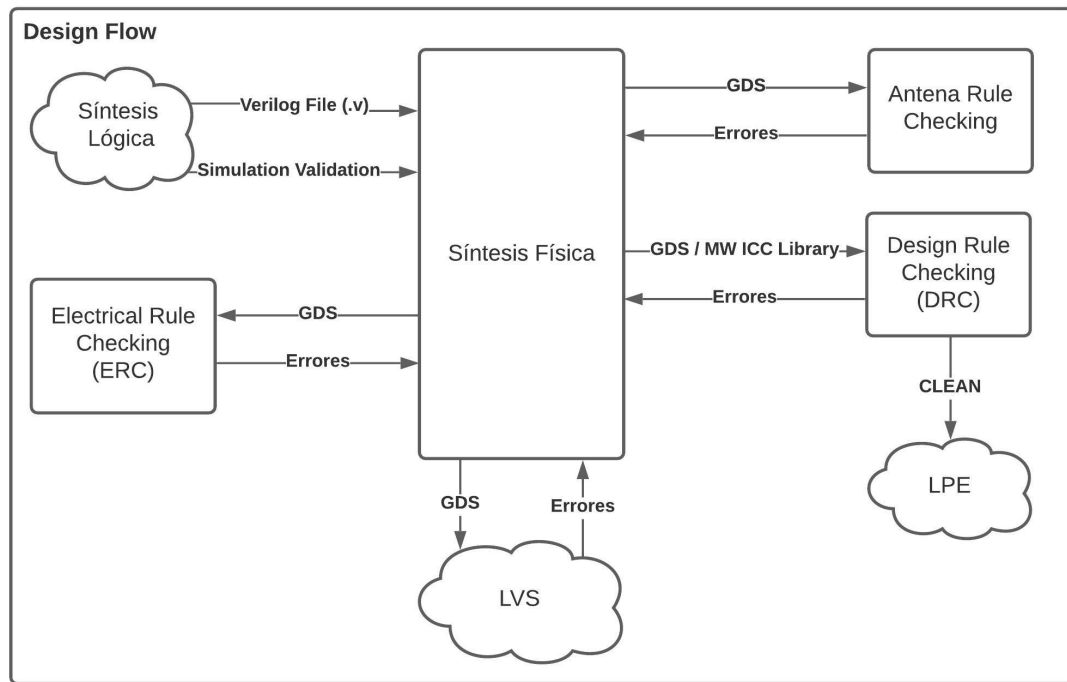


Figura 3: Input Output design flow diagram

## 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] M. Flores, “Corrección de anillo de entradas/salidas y pruebas de antenna y ERC para la definición del flujo de diseño del primer chip con tecnología nanométrica desarrollado en Guatemala,” en *Trabajo de Graducacion, Modalidad Tesis*, Facultad de Ingeniería Universidad del Valle de Guatemala, 2020.
- [5] *EDA101 - Introduction to Electronic Design Automation*. Cadence Design Systems, abr. de 2020. dirección: <https://www.youtube.com/watch?v=HkT363NU0BA>.
- [6] Synopsys. dirección: <https://www.synopsys.com/silicon-design.html>.
- [7] 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.
- [8] Synopsys. dirección: <https://www.synopsys.com/implementation-and-signoff/physical-implementation/ic-compiler.html>.
- [9] —, dirección: <https://www.synopsys.com/silicon/mask-synthesis/icv-workbench.html>.