

2015-12

# Diseño del módulo transmisor serial de datos de sistema SerDes para Protocolo PCI express 1

López-Félix, Carlos C.

---

López-Félix, C. C. (2015). Diseño del módulo transmisor serial de datos de sistema SerDes para Protocolo PCI express 1. Trabajo de obtención de grado, Especialidad en Diseño de Sistemas en Chip. Tlaquepaque, Jalisco: ITESO.

Enlace directo al documento: <http://hdl.handle.net/11117/3751>

*Este documento obtenido del Repositorio Institucional del Instituto Tecnológico y de Estudios Superiores de Occidente se pone a disposición general bajo los términos y condiciones de la siguiente licencia:*

*<http://quijote.biblio.iteso.mx/licencias/CC-BY-NC-ND-2.5-MX.pdf>*

*(El documento empieza en la siguiente página)*

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE OCCIDENTE**

**Especialidad en Diseño de Sistemas en Chip**

Reconocimiento de Validez Oficial de Estudios de nivel superior según Acuerdo Secretarial 15018,  
publicado en el *Diario Oficial de la Federación* el 29 de noviembre de 1976

**DEPARTAMENTO DE ELECTRÓNICA, SISTEMAS E INFORMÁTICA**



**Diseño del módulo Transmisor Serial de Datos de Sistema  
SerDes para Protocolo PCI express 1**

Tesina para obtener el grado de:

**Especialista en diseño de sistemas en chip**

Presenta

Carlos César López Félix:

Directores de tesina: Esteban Martínez Guerrero y Federico Lobato López:

Guadalajara, Jalisco, Diciembre 2015



## Agradecimientos

El autor de esta tesina agradece muy profundamente a sus padres por haberle dado la oportunidad a través de la vida y el apoyo en los estudios universitarios, el poder superarse cada día un poco más con estudios de posgrado en áreas tan importantes como lo es la especialidad de diseño de sistemas en Chip.

A su esposa Amalia Araiza e hijos Sergio Alberto y Karla Gabriela por su apoyo incondicional y ejemplo de vida y superación, ha tratado de ser un ejemplo y guía para ellos, pero en muchas ocasiones él más bien toma su ejemplo para inspirarse y tener el empuje para seguir con su preparación.

A sus hermanas Patricia e Hilda las cuales son un apoyo importante por su compañía y ejemplo de vida.

A su hermano Sergio que aunque hace mucho tiempo nos dejó, son muy importantes todos los momentos buenos y malos vividos a su lado.

A su familia en general por sentir siempre su compañía y apoyo.

Al Doctor Esteban Martínez y Federico Lobato por su apoyo y orientación en todo el proceso de la especialidad de diseño de sistemas en Chip.

A los maestros que se tuvo durante el desarrollo de los estudios, porque de ellos aprendió todo lo referente a la especialidad.

A PNPC-CONACYT por el gran soporte económico que aportó para realizar los estudios.

Al ITESO por ser su alma mater y guía en su superación continua.



# Abstract

Serial communication protocols are developing and growing faster than any other communications protocols; this due to the economic advantages they offer because they use less communications channels, also the use of differential pairing techniques using two complementary signals that helps reduce the electromagnetic emissions and susceptibility, however, their complexity is higher than other available communications protocols.

This technical report documents the design process of a serial transmitter with amplitude, pre-emphasis and programmable impedance coupling, the topology used for this implementation is the segmented self-series terminated transmitters (SSSTT) which manage the functions just described above, it does this efficiently, with low area use and lower power consumption than other conventional structures. The technology used is IBM 180nm CMOS technology (process IBM cmrf7sf) with MOSIS license.

The process of design is from bottom-up methodology, because all the cells used are composed from repetitive basic cells; schematic designs were developed and tested first from basic to complex cells, using test benches.

The layout of each cell is full custom; no standard cells were used because this is an analog design, despite the digital cell composition of it. For this first approach, only LVS tests were performed because individual cells are going to change their initial topology at the moment they are initially placed until the final layout site is determined, complex cells are built for the complete application.

The final result was the design of a transmitter cell for one channel (+Tx) at schematic and layout level, verified with LVS test; DRC and other post layout evaluations will be performed during next design stage due to time constrains. The other channel (-Tx) is the duplication of channel +Tx.

# Índice

Agradecimientos .....	iii
Abstract.....	iv
Lista de Figuras .....	ix
Lista de Tablas .....	xii
1 Introducción .....	1
2 Descripción del sistema de serialización-deserialización (SerDes) .....	4
2.1 Descripción General del Serializador .....	4
2.1.1 Etapa analógica de recepción .....	5
2.1.2 Módulo digital de de-serialización .....	5
2.1.3 Módulo digital de serialización .....	5
2.1.4 Etapa analógica de transmisión .....	6
2.1.5 Módulo de pruebas .....	6
2.2 Información general del protocolo PCI Express 1 .....	6
2.2.1 Sistema General .....	6
2.2.2 Parámetros de Diseño .....	7
2.2.3 Descripción de Señales .....	7
2.2.4 Características Generales y Especificaciones Eléctricas .....	8
2.2.5 Requerimientos del Reloj .....	8
2.2.6 Requerimientos de Lógica Digital .....	10
2.2.7 Inicialización de la Aplicación e Información del Reset.....	12
2.2.8 2.5 GT/s Compliance Pattern.....	13
2.3 Objetivo del Proyecto .....	14
2.4 Consideraciones.....	15
2.5 Restricciones.....	16
3 Diseño esquemático del módulo Transmisor de datos del sistema SerDes .....	18

3.1	Desarrollo del Módulo transmisor.....	19
3.1.1	Diseño del circuito Inversor.....	19
3.1.2	Diseño de compuerta NOR de dos entradas (NOR2).....	20
3.1.3	Diseño de NAND de dos entradas (NAND2).....	21
3.1.4	Diseño compuerta NOR de tres entradas (NOR3).....	21
3.1.5	Diseño compuerta NAND de tres entradas (NAND3).....	22
3.1.6	Diseño del Buffer de tres estados.....	23
3.1.7	Diseño del Inversor-Retardador.....	24
3.1.8	Diseño del decodificador de 4 entradas y 12 salidas, (DECO 4 a 12)....	25
3.1.9	Diseño del decodificador de 2 entradas y 3 salidas, (DECO 2 a 3).....	28
3.1.10	Diseño del Multiplexor (MUX 2 a 1).....	28
3.1.11	Diseño del circuito amarrador (tied).....	29
3.1.12	Diseño de la unidad de control ZAP.....	30
3.1.13	Circuito Transmisor de datos.....	31
4	Prueba y Verificación del Módulo Transmisor.....	33
4.1	Setup del Control de Impedancias del Transmisor:.....	33
4.1.1	Pruebas del control impedancia, condición de alta impedancia:.....	36
4.1.2	Pruebas del control impedancia, condición de impedancia intermedia: .	37
4.1.3	Pruebas del control impedancia, condición de baja impedancia:.....	38
4.2	Setup del Control de Amplitud del Transmisor:.....	38
4.2.1	Pruebas del control de amplitud:.....	40
4.3	Setup del Control de Pre Énfasis del Transmisor:.....	40
4.3.1	Pruebas del control de pre énfasis:.....	42
4.4	Setup para probar la señal Random a través del Transmisor:.....	42
4.4.1	Programa Generador de señal Random:.....	43
4.4.2	Pruebas del circuito Inversor.....	46
4.4.3	Pruebas de compuerta NOR de dos entradas (NOR2).....	48
4.4.4	Pruebas de NAND de dos entradas (NAND2).....	49
4.4.5	Pruebas compuerta NOR de tres entradas (NOR3).....	51

4.4.6	Pruebas compuerta NAND de tres entradas (NAND3) .....	52
4.4.7	Pruebas del Buffer de tres estados .....	53
4.4.8	Pruebas del Inversor-Retardador .....	54
4.4.9	Pruebas del decodificador de 4 entradas y 12 salidas, (DECO 4 a 12) ..	55
4.4.10	Pruebas del decodificador de 2 entradas y 3 salidas, (DECO 2 a 3) .....	57
4.4.11	Pruebas del Multiplexor (Mux 2 a 1) .....	58
4.4.12	Pruebas del circuito amarrador (tied) .....	59
4.4.13	Pruebas de la unidad de control ZAP .....	61
5	Diseño del Layout del módulo Transmisor de datos del sistema SerDes .....	66
5.1	Primera Aproximación del Floorplan .....	67
5.2	Diseño de layout de celdas básicas del módulo Transmisor .....	69
5.2.1	Layout del circuito Inversor: .....	70
5.2.2	Layout de compuerta NOR de dos entradas (NOR2) .....	71
5.2.3	Layout de la compuerta NAND de dos entradas, (NAND2) .....	72
5.2.4	Layout de la compuerta NOR de tres entradas (NOR3) .....	73
5.2.5	Layout de la compuerta NAND de tres entradas (NAND3) .....	74
5.2.6	Layout del circuito amarrador (tied). .....	75
5.2.7	Layout del Buffer de tres estados .....	75
5.2.8	Layout del Inversor-Retardador .....	77
5.2.9	Layout del Multiplexor .....	78
5.2.10	Layout del circuito Buffer .....	79
5.3	Layout completo del módulo transmisor .....	79
5.3.1	Buffer de Tres Estados .....	80
5.3.2	Decodificador 2 Entradas 3 Salidas .....	80
5.3.3	Decodificador 4 Entradas 12 Salidas .....	81
5.3.4	Unidad ZAP .....	81
5.3.5	Layout de un canal de módulo de Transmisión .....	82
6	Conclusiones .....	85
7	Lista de Referencias .....	86

8	Apéndice.....	88
8.1	Ecuación del Divisor de Voltaje.....	88

## Lista de Figuras

Figura 2-1: Sistema SerDes [2] .....	4
Figura 2-2: Arquitectura del protocolo PCI Express[4] .....	7
Figura 2-3: Capa física en el protocolo PCI Express [4].....	8
Figura 2-4: Driver en el protocolo PCI Express [4] .....	8
Figura 2-5: Capacitor de desacople en el driver del protocolo PCI Express [4] .....	9
Figura 2-6: Transmisión de datos en el protocolo “PCI Express” [4] .....	11
Figura 2-7: “PCI Express System Architecture” [4].....	13
Figura 2-8: 2.5GT/s compliance pattern [4] .....	13
Figura 2-9: Hardware de prueba de compliance [4] .....	14
Figura 2-10: Diagrama de ojo durante la prueba de medición de jitter [4].....	14
Figura 3-1: Diagrama a bloques del circuito transmisor de datos [5] .....	18
Figura 3-2: Esquemático del circuito inversor [5].....	20
Figura 3-3: Esquemático del circuito de compuerta NOR2 [5] .....	20
Figura 3-4: Esquemático del circuito de compuerta NAND2 [5] .....	21
Figura 3-5: Esquemático del circuito de compuerta NOR3 [5] .....	22
Figura 3-6: Esquemático del circuito de compuerta NAND3 [5] .....	23
Figura 3-7: Esquemático del circuito Buffer de tres estados [5] .....	24
Figura 3-8: Esquemático del circuito Retardador-Inversor [5] .....	25
Figura 3-9: Tres bloques integran el decodificador de 4 a 12 [5] .....	26
Figura 3-10: Diseño interno del bloque 1 [5].....	26
Figura 3-11: Diseño interno del bloque 2 [5].....	27
Figura 3-12: Diseño interno del bloque 3 [5].....	27
Figura 3-13: Decodificador 2 entradas 3 salidas [5] .....	28
Figura 3-14: Esquemático del circuito Mux 2 a 1 [5].....	29
Figura 3-15: esquemático del circuito Amarrador (Tied) [5] .....	30
Figura 3-16: Diagrama a bloques de la unidad ZAP [5].....	30
Figura 3-17: Circuito de celda básica de la unidad ZAP [5].....	31
Figura 3-18: Esquemático test bench del módulo transmisor de datos [5].....	31
Figura 4-1: Combinación digital a nivel señal de entradas.....	35
Figura 4-2: Verificación de cambio de impedancia (Impedancia Alta).....	36
Figura 4-3: Verificación de cambio de impedancia (Impedancia Intermedia).....	37
Figura 4-4: Verificación de cambio de impedancia (Impedancia Baja).....	38

Figura 4-5: Combinación nivel señal de fuentes de prueba Amp_A y Amp_B .....	39
Figura 4-6: Verificación de control de amplitud .....	40
Figura 4-7: Combinación a nivel señal de fuentes Eq_a y Eq_B.....	41
Figura 4-8: Verificación de pre-énfasis .....	42
Figura 4-9: Diagrama Test Bench para el inversor .....	47
Figura 4-10: Respuesta del Inversor .....	47
Figura 4-11: Test bench del circuito de compuerta NOR2 .....	48
Figura 4-12: Respuesta transitoria del circuito de Compuerta NOR2 .....	49
Figura 4-13: Test Bench para simular el circuito de compuerta NAND2 .....	50
Figura 4-14: Respuesta transitoria del circuito de compuerta NAND2 .....	50
Figura 4-15: Test bench para simular el circuito de compuerta NOR3.....	51
Figura 4-16: Respuesta transitoria del circuito de compuerta NOR3 .....	51
Figura 4-17: Test bench para simular el circuito de compuerta NAND3 .....	52
Figura 4-18: Respuesta de compuerta NAND de tres entradas .....	52
Figura 4-19: Test bench del circuito Buffer de tres estados .....	53
Figura 4-20: Respuesta transitoria del circuito Buffer de tres estados .....	53
Figura 4-21: Test bench del circuito Retardador-Inversor .....	54
Figura 4-22: Respuesta del Buffer Retardador-Inversor.....	55
Figura 4-23: Test Bench para decodificador de 4 a 12.....	56
Figura 4-24: Respuesta del Decodificador 4 entradas 12 salidas .....	56
Figura 4-25: Decodificador 2 entradas 3 salidas test bench.....	57
Figura 4-26: Respuesta del decodificador 2 entradas 3 salidas.....	57
Figura 4-27: Test-bench del circuito Mux 2 a 1. ....	58
Figura 4-28: Respuesta transitoria del Mux 2 a 1 .....	59
Figura 4-29: Test bench del circuito Amarrador (Tied) .....	60
Figura 4-30: Respuesta transitoria del circuito Amarrador (Tied).....	60
Figura 4-31: Test bench de la unidad de control ZAP .....	61
Figura 4-32: Evaluación de cambio de amplitud.....	62
Figura 4-33: Evaluación de cambio de impedancia.....	62
Figura 4-34: Evaluación de pre-énfasis. ....	63
Figura 5-1: Diagrama a bloques del circuito transmisor de datos [5]. ....	66
Figura 5-2: Layout preliminar del transmisor. ....	69
Figura 5-3: Layout para el inversor.....	70
Figura 5-4: Resultados de la prueba <b>Assura</b> -LVS de la celda inversor. ....	71
Figura 5-5: Layout compuerta NOR.....	71
Figura 5-6: Compuerta NAND .....	72
Figura 5-7: Compuerta NOR de tres entradas.....	73

Figura 5-8: Compuerta NAND de tres entradas. ....	74
Figura 5-9: Circuito Amarrador (Tied).....	75
Figura 5-10: Buffer de tres estados. ....	76
Figura 5-11: Buffer Retardador-Inversor.....	77
Figura 5-12: Multiplexor de 2 entradas.....	78
Figura 5-13: Layout del circuito Buffer.....	79
Figura 5-14: Buffer de tres estados con dos resistencias de 820 $\Omega$ .....	80
Figura 5-15: Codificador de 2 a 3 tipo termómetro.....	80
Figura 5-16: Codificador de 4 a 12 tipo termómetro.....	81
Figura 5-17: Unidad básica de control ZAP.....	81
Figura 5-18: Diseño completo canal +TX.....	82



## Lista de Tablas

Tabla 2-1: Paquete de Información Protocolo PCI Express [4] .....	7
Tabla 3-1: Dimensiones de los transistores del circuito compuerta NOR2 .....	20
Tabla 3-2: Dimensiones de los transistores del circuito de compuerta NAND2 .....	21
Tabla 3-3: Dimensiones de los transistores del circuito de compuerta NOR3 .....	21
Tabla 3-4: Dimensiones de los transistores del circuito de compuerta NAND3 .....	22
Tabla 3-5: Dimensiones de los transistores del circuito Buffer de tres estados .....	23
Tabla 3-6: Dimensiones de los transistores del circuito Retardador Inversor .....	25
Tabla 3-7: Dimensiones de los transistores del Circuito MUX 2 a 1.....	28
Tabla 3-8: Dimensiones de los transistores del Circuito Tied .....	29
Tabla 4-1: Configuración de fuentes de Amplitud y Pre énfasis.....	33
Tabla 4-2: Configuración de fuentes de entrada .....	34
Tabla 4-3: Combinación digital numérica de entradas .....	34
Tabla 4-4: Configuración señal Data .....	35
Tabla 4-5: Configuración de fuentes de Impedancia y Pre énfasis.....	38
Tabla 4-6: Configuración de fuentes de prueba Amp_A y Amp_B.....	39
Tabla 4-7: Combinación numérica de fuentes de prueba Amp_A y Amp_B .....	39
Tabla 4-8: Configuración de fuentes de Impedancia y Amplitud. ....	40
Tabla 4-9: Configuración de fuentes de entrada Eq_A y Eq_B para la evaluación.....	41
Tabla 4-10: Combinación numérica de fuentes Eq_a y Eq_B.....	41
Tabla 4-11: Configuración de fuentes de Impedancia, Amplitud y Ecuación.....	43
Tabla 5-1: Dimensiones de los bloques funcionales. ....	67
Tabla 5-2: Dimensión bloque de unidad ZAP.....	67
Tabla 5-3: Dimensión bloque decodificador 4-12. ....	67
Tabla 5-4: Dimensión bloque decodificador 2-3. ....	68
Tabla 5-5: Dimensión bloque del circuito total de 1 canal. ....	68
Tabla 5-6: Dimensión bloque del circuito total de 1 canal Invertido. ....	68
Tabla 5-7: Dimensión bloque completo del Transmisor. ....	68
Tabla 5-8: Dimensiones circuito inversor .....	70
Tabla 5-9: Dimensiones Compuerta NOR.....	71
Tabla 5-10: Dimensiones Compuerta NAND.....	72
Tabla 5-11: Dimensiones Compuerta NOR3.....	73
Tabla 5-12: Dimensiones Compuerta NAND3.....	74

Tabla 5-13: Dimensiones Circuito Tied.....	75
Tabla 5-14: Dimensiones Buffer de tres estados.....	76
Tabla 5-15: Dimensiones Retardador Inversor.....	77
Tabla 5-16: Dimensiones Circuito MUX.....	78
Tabla 5-17: Dimensiones Circuito Buffer .....	79

# 1 Introducción

Este reporte técnico presenta el diseño de un transmisor para información digital serial de manera diferencial, los canales de transmisión son +Tx y –Tx, uno es complemento del otro, este desarrollo es parte de un proyecto de diseño de un sistema SerDes (Serializador-Deserializador) para protocolo de comunicación serial PCI Express 1 hasta 1.5 GHz en tecnología CMOS de 180 nm proceso IBM cmrf7sf con licencia de MOSIS.

Se basa en una topología self-series terminated transmitter (SSSTT) el cual tiene la capacidad de manipular la amplitud, pre-énfasis e impedancia del acople con la línea de transmisión de datos seriales para optimizar la transferencia de información digital. Esta configuración de parámetros de transmisión de datos puede hacerse de manera dinámica durante el proceso de comunicación si se implementa una máquina de estados que controle los bits de configuración de los mismos, dicha implementación queda fuera del alcance de este proyecto y se limita solo a hacer una calibración inicial del sistema antes de comenzar con el proceso de comunicación.

La comunicación serial de datos es un sistema que beneficia al mundo de la tecnología digital al requerir menos canales de comunicación, como ocurre con los protocolos de comunicación en paralelo, además de evitar interferencia con los canales de comunicación adyacentes, ya que al utilizarse señales complementarias (+Tx y –Tx) evitan interferencia entre si y evitan también emisiones electromagnéticas con el entorno al cancelarse sus componentes, al mismo tiempo hace que estos canales de comunicación sean más inmunes a interferencia electromagnética que provenga de otros dispositivos en el entorno del sistema, también tiene ventajas en el alcance ya que puede cubrir distancias más largas sin problemas de degradación de señal. Tiene la desventaja de que su implementación es más compleja que los sistemas paralelos de transmisión de datos u otros protocolos existentes.

El desarrollo de este sistema de comunicación serial es de carácter formativo como parte de la especialidad de diseño de sistemas en Chip y tiene el objetivo de ilustrarnos el proceso completo para diseñar y mandar fabricar circuitos integrados de aplicación específica. Es de dominio estudiantil sus resultados y posibles mejoras en el futuro que se puedan agregar.

La meta planteada al inicio del proyecto era mandar fabricar el sistema SerDes desarrollado por el equipo de estudios con MOSIS pero limitantes con el tiempo y

algunas herramientas del software **Cadence** nos impidieron alcanzar este objetivo. Aun así, este reporte técnico ofrece un buen desarrollo del diseño del módulo del transmisor del sistema SerDes y lo lleva hasta el nivel de layout como una primera aproximación del mismo.

Pruebas de post layout faltantes podrán desarrollarse en una siguiente etapa con el objetivo de fabricar un prototipo del proyecto.

El capítulo 2 da una breve descripción del protocolo de comunicación PCI Express 1, así como algunos de sus parámetros importantes y sugerencias de evaluación del mismo.

El capítulo 3 describe el desarrollo del transmisor a nivel esquemático, se mencionan las consideraciones tomadas para el dimensionamiento de los transistores de las celdas básicas y de la agrupación de las mismas en celdas más complejas que constituyen el transmisor.

El capítulo 4 describe los test bench utilizados para las evaluaciones de las celdas, el proceso que se siguió en las evaluaciones de las mismas y los resultados gráficos y numéricos obtenidos.

El capítulo 5 muestra el resultado de la generación del layout de las celdas básicas, una primera aproximación al área de silicio requerida para el circuito transmisor y la conjunción de estas celdas básicas en celdas más complejas hasta obtener el layout del transmisor completo para un solo canal, el segundo canal es una duplicación del canal desarrollado.



## 2 Descripción del sistema de serialización-deserialización (SerDes)

### 2.1 Descripción General del Serializador

Un sistema de comunicación digital SerDes (serializador-deserializador) serializa y deserializa información digital para transmitir datos con un número de canales de comunicación inferior al número de bits por paquete de datos que se están recibiendo [1]. Generalmente son uno o dos cables de transmisión de datos (*Tx*) y uno o dos cables para la recepción (*Rx*). Utiliza una codificación 8b10b que es una interfaz serial que usa dos pares de señales LVDS [1]. La interfaz básica LVDS es un enlace diferencial simple en una o dos direcciones. Cada enlace requiere un resistor de  $100\Omega$  ( $\pm 15\%$ ) como terminación de impedancia en cada uno de los extremos de recepción [1].

La Figura 2-1, presenta el diagrama a bloques del sistema SerDes de **Freescalle**, que se propuso desarrollar en esta especialidad, consiste en una etapa de recepción analógica de señal, sistema digital de des-serIALIZACIÓN, sistema digital de serialización, etapa de transmisión analógica y un módulo de pruebas para hacer verificaciones iniciales de hardware [2].

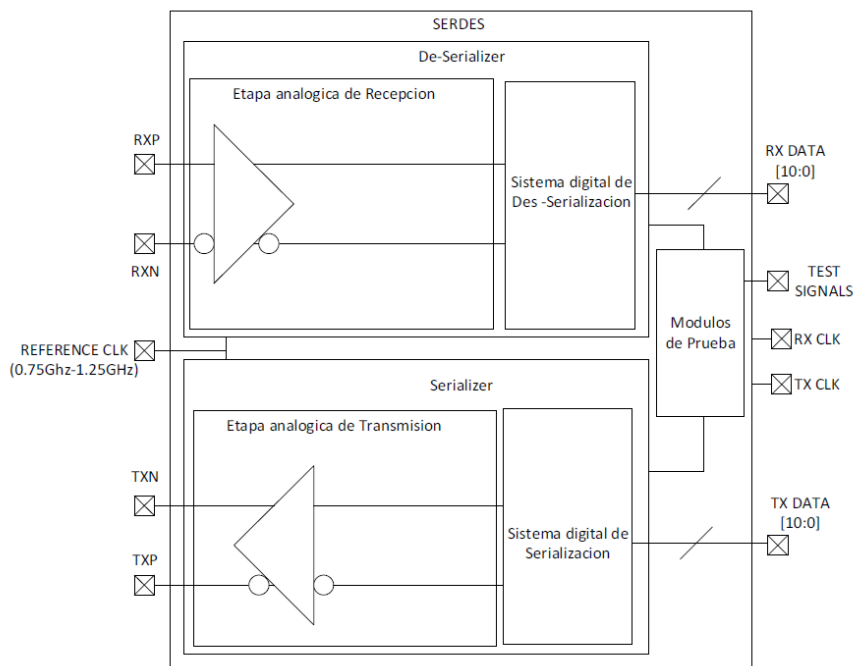


Figura 2-1: Sistema SerDes [2]

### 2.1.1 Etapa analógica de recepción

Recibe una señal digital diferencial (RXP, RXN) y debe proveer una señal CMOS con el menor ruido y jitter posible para ser utilizada por las etapas digitales. La etapa analógica de recepción tiene como objetivo el compensar la atenuación en amplitud experimentada por los datos seriales al ser transmitidos a través del canal de comunicación. El sistema debe cumplir con las especificaciones de velocidad y contar con un circuito de polarización a fin de lograr la compensación a través de las diferentes esquinas de proceso, voltaje y temperatura (PVT's) [1].

### 2.1.2 Módulo digital de de-serialización

Este módulo es el encargado de recibir el patrón digital de datos y sincronizarlo con el reloj de referencia. Para este propósito se utiliza un circuito de recuperación de datos (*CDR por sus siglas en inglés*). Para el *CDR* se pueden utilizar varias técnicas, muchas están basadas en la toma de muestras a una frecuencia más alta que el bit rate del dato.

Una vez recuperado el dato y sincronizado con el reloj se ejecuta la conversión digital serie paralelo. El reloj de recepción (*RX\_CLK*) debe estar sincronizado con el dato serial recuperado e indica cuando se debe tomar el dato paralelo (*RX\_DATA*). Este módulo debe poder seleccionar la entrada de datos serial analógica (modo de aplicación) o una entrada de datos serial digital proveniente del módulo de transmisión digital para el modo BIST.

El dato serial digital proveniente del transmisor debe pasar también por el proceso *CDR* y conversión a dato paralelo. Este módulo debe contener un bloque que genere el mismo dato paralelo interno generado en el transmisor y lo compare con el dato paralelo obtenido de la conversión serie-paralelo. Una señal de salida debe indicar si hay un error o discrepancia entre los datos. Para realizar la prueba de BIST se tiene que crear una secuencia de inicio, sincronía y comparación que se pueda controlar externamente [3].

### 2.1.3 Módulo digital de serialización

1. El reloj de transmisión debe ser generado para sincronizar las tomas de muestras del dato paralelo (*TX\_DATA*).
2. El dato paralelo es almacenado y sincronizado con el reloj de transmisión para posteriormente ser convertido a un dato serial.
3. Este dato serial se debe proveer a la etapa de transmisión analógica para enviar el dato serial fuera del chip.
4. En este módulo se debe incluir un generador de datos paralelos interno para realizar BIST. Por medio de señales de control se debe poder seleccionar el dato paralelo

externo e interno. El dato paralelo interno se utilizara durante BIST y se convertirá a serial con el mismo módulo utilizado en la aplicación con el dato externo. Este dato serial se debe enviar a la parte digital de recepción (creando un “*loopback*” con el sistema digital de des-serialización) [3].

#### *2.1.4 Etapa analógica de transmisión*

Esta etapa se encarga de transmitir los datos diferenciales a través de una línea de transmisión que tiene una impedancia de 100  $\Omega$ . Sistemas avanzados de transmisión de datos tienen la capacidad de hacer procesos de pre-énfasis en los datos transmitidos, cambios en la amplitud de la señal transmitida y cambios en la impedancia de acoplamiento a la línea de transmisión para optimizar el envío de datos digitales [3].

#### *2.1.5 Módulo de pruebas*

La etapa de integración del sistema tiene como objetivo crear la lógica y estructuras que permitan la comunicación de las diferentes etapas. El sistema completo debe contar con la capacidad de poder probar los bloques digitales de manera individual y en conjunto fuera del sistema utilizando diferentes técnicas de prueba como son BIST y SCAN [1].

## **2.2 Información general del protocolo PCI Express 1**

### *2.2.1 Sistema General*

1. El patrón de datos serial debe ser en código 8b10b para mantener un balance en DC.
2. La velocidad del patrón de datos es de 1.5Gbps a 2.5Gbps.
3. La frecuencia del reloj de referencia es de 0.75Ghz a 1.25Ghz para poder soportar la frecuencia de los datos.
4. La velocidad de los relojes de recepción y transmisión es 10 veces la de la frecuencia del dato.
5. Debe integrar un generador de patrón de datos interno para realizar built-in self-test (BIST).
6. Todos los módulos deben tener la opción de “bypass” con motivos de verificación. Esto es, por medio de señales de control, poder dejar pasar los datos de cada módulo sin ningún procesamiento ni adecuación al módulo siguiente.
7. El sistema debe permitir la conexión interna del dato paralelo obtenido por el deserializer hacia el serializer para poder crear un lazo interno que permita convertir el mismo dato nuevamente a un patrón serial por el transmisor, con motivos de prueba [3].



### 2.2.2 Parámetros de Diseño

El paquete de información del protocolo PCI Express es como sigue: se encuentra compuesto por un inicio de trama (*SOF, Star Of Frame*), un número de secuencia (*SEQ#, Sequence Number*), un encabezado (*Header*), un campo de datos (*Data Field*), un control de redundancia cíclica (*CRC, Cyclic Redundancy Check*) y un final de trama (*EOF, End Of Frame*) [4].

Inicio de Trama SOF	Numero de secuencia SEQ#	Header	Campo de Datos Data	Control de Redundancia Cíclica, CRC	Fin de Trama EOF
------------------------	-----------------------------	--------	------------------------	--	---------------------

Tabla 2-1: Paquete de Información Protocolo PCI Express [4]

### 2.2.3 Descripción de Señales

La lógica de transmisión de la *capa física* procesa paquetes de información que llegan del *Data Link Layer*, luego los convierte en un *stream* de datos seriales. El *stream* de bits se envía a una razón de 2.5 Gbits/s hacia el canal de transmisión.

El receptor recibe el *stream* serial de bits con una señal de reloj embebida que es recuperado y lo convierte en un conjunto de datos en paralelo que son símbolos; procesa los símbolos, ensambla los paquetes de información y los envía al *Data Link Layer* [4].

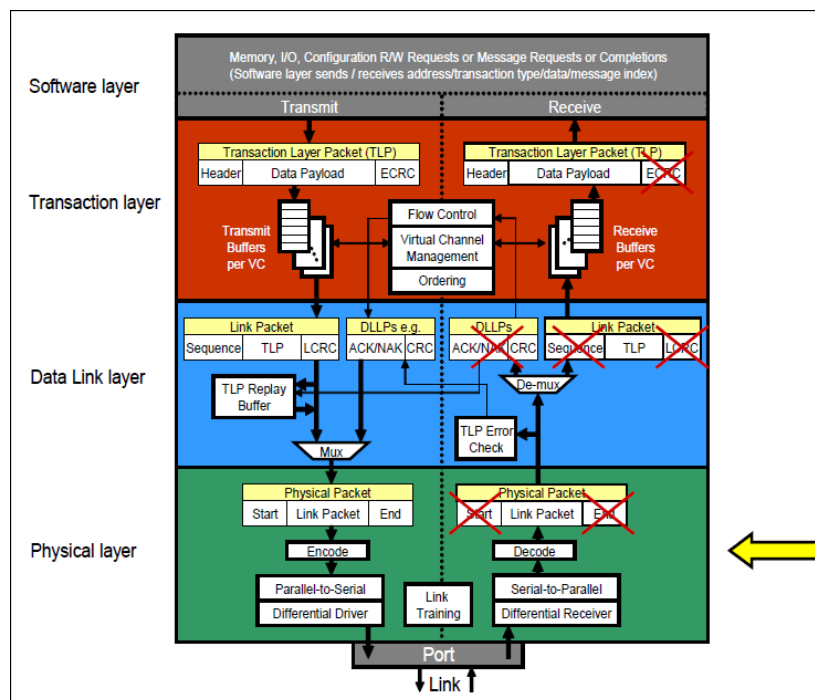


Figura 2-2: Arquitectura del protocolo PCI Express[4]

Existen dos bloques que conforman el *Layer Físico*, estos son el *Layer Físico* de la Lógica Digital y el *Layer Físico* Eléctrico [4].

### 2.2.4 Características Generales y Especificaciones Eléctricas

Este block contiene drivers diferenciales (transmisores) y receptores diferenciales de señal digital. El transmisor serializa la información y convierte el *stream* de datos en señal eléctrica que tiene un reloj embebido. El receptor detecta la señal eléctrica y genera el *stream* de señal serial para de-serializarla en información, y provee esta información a la lógica digital de la capa física junto con el reloj recuperado [4].

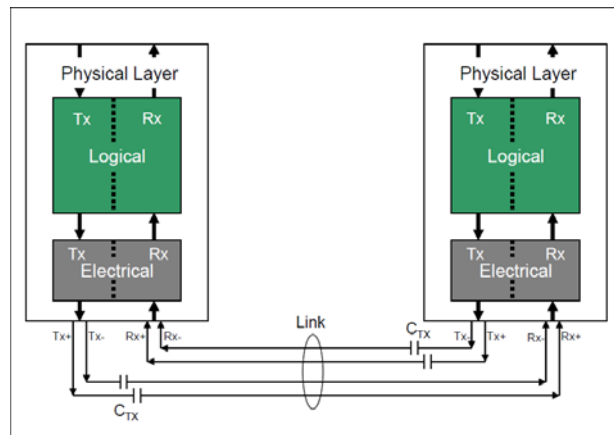


Figura 2-3: Capa física en el protocolo PCI Express [4]

Las señales eléctricas LVDS (Low Voltage Differential Signal) de alta velocidad son usadas en las implementaciones de drivers y receptores [4].

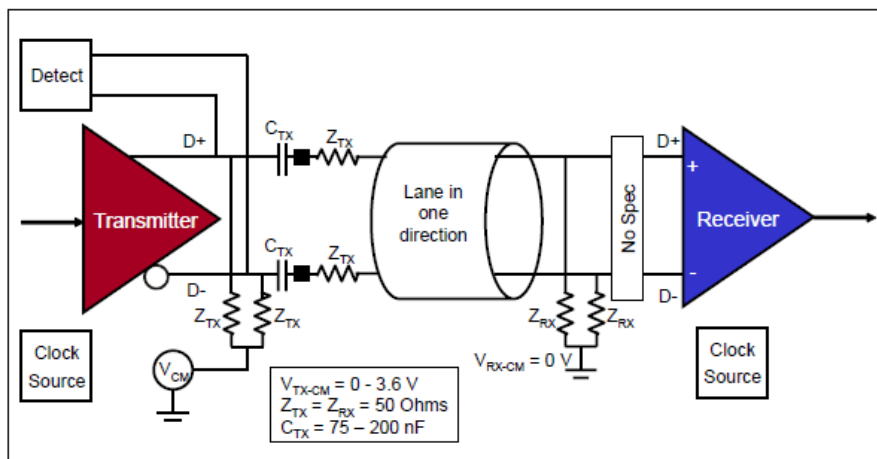


Figura 2-4: Driver en el protocolo PCI Express [4]

### 2.2.5 Requerimientos del Reloj

El transmisor envía datos a 2.5 Gbits/s. La precisión requerida es de  $\pm 300$  ppm de la frecuencia central. El máximo *skew* permitido es de una señal de reloj cada 1666 pulsos de reloj. La máxima desviación de fase entre el receptor y el transmisor es de 600 ppm [4].

*Spread Spectrum Clocking (SSC)* es una técnica usada para modular la frecuencia de reloj lentamente para reducir las radiaciones de EMC a la frecuencia central.

La característica de impedancia en el canal de comunicación es de  $100\Omega$  diferencial (nominal), y si es *single-ended* DC en modo común de impedancia es de  $50\Omega$ . Esta impedancia debe coincidir en el transmisor y el receptor.

El transmisor debe cumplir con los parámetros del estándar  $Z_{tx-diff-dc}$  cada vez que se transmite señal diferencial durante el estado de encendido full-on L0.

Cuando una señal diferencial no es transmitida, el transmisor debe mantener su salida de impedancia al mínimo  $Z_{tx-dc}$  que es de  $40\Omega$ , pero también podría poner el driver en alta impedancia. Esto puede ayudar a reducir consumo de potencia cuando esta no es requerida.

El receptor debe cumplir con el estándar  $Z_{rx-diff-dc}$  de  $100\Omega$  en cualquier momento en que la señal diferencial es transmitida durante el estado de encendido full-on L0, al igual que en cualquier modo de bajo consumo siempre y cuando el dispositivo recibe una polarización adecuada.

El voltaje de DC de modo común del transmisor  $V_{tx-dc-cm}$  debe permanecer con un voltaje estable. Este solo se apaga cuando el transmisor es puesto en el modo de bajo consumo L2 o L3, durante el cual la alimentación es removida. El diseñador puede escoger el voltaje en un rango de 0 a 3.6 V

El receptor es des-acoplado del transmisor con un capacitor. Esto permite al receptor tener su propio voltaje DC de modo común. Este voltaje se especifica a 0V. No es claro si esto debe cumplirse, no es requerido que este voltaje sea de 0V a la entrada del amplificador diferencial del receptor [4].

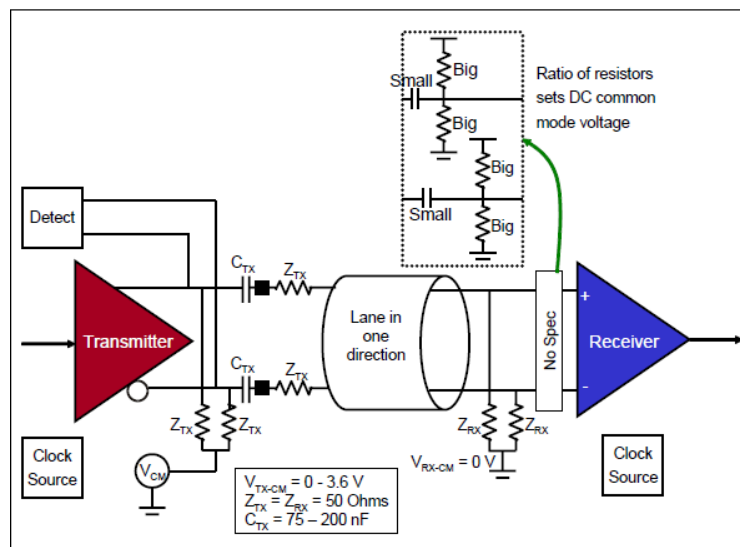


Figura 2-5: Capacitor de desacople en el driver del protocolo PCI Express [4]

El dispositivo debe cumplir con requerimientos de ESD y de corto circuito en todos los pines del dispositivo, véase el estándar JEDEC JESE22-A114-A [4].

### 2.2.6 Requerimientos de Lógica Digital

Los elementos que conforman la lógica digital de transmisión son:

- Multiplexor (MUX)
- Byte striping logic (solo necesaria si el enlace de transmisión será de más de una línea de datos)
- Scramblers
- Codificador 8b/10b
- Convertidor Paralelo a Serial.

TLP y DLLP que vienen del *Data Link Layer* son empacados en el buffer del transmisor. Con la ayuda del Multiplexor, el *Layer Físico* trasfiere a *Frames* los TLP o los DLLP con caracteres de *Start* y *End*. Estos caracteres son utilizados por el receptor para localizar el inicio y final de datos recibidos.

Los paquetes convertidos en *Frames* son enviados al *Byte Striping logic* que multiplexa los bytes hacia las líneas. One byte del paquete es transferido a una línea, el siguiente byte al siguiente *Lane* disponible, y así sucesivamente para los *Lanes* disponibles.

El *Scrambler* usa un algoritmo para pseudo-aleatoriamente revolver cada byte del paquete de datos. El *Start* y *End* del *Frame* no son revueltos. *Scrambling* elimina los patrones repetitivos de un bit *stream*.

Los 8 bits revueltos (8b) son codificados en un símbolo de 10 bits (10b) por la lógica 8b/10b. Existe una pérdida de 25% en el performance de la línea de transmisión debido a la expansión de cada byte en un carácter de 10 bits. El propósito general de este proceso es de que sea más fácil de ser recuperado el reloj que está incluido en el *streaming* de datos.

El símbolo de 10 bits es convertido a un *stream* de bits seriales por el convertidor de Paralelo a Serie. Esta lógica usa un reloj de 2.5 GHz para enviar los paquetes de datos por cada *Lane*. Los paquetes de bits seriales son enviados al bloque eléctrico que transmite la señal de manera diferencial [4].

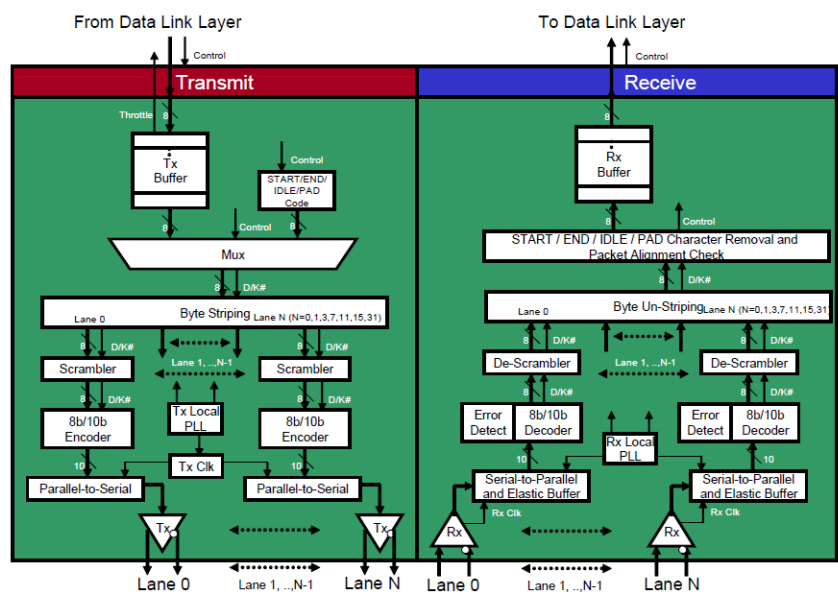


Figura 2-6: Transmisión de datos en el protocolo "PCI Express" [4]

Los elementos que conforman la lógica de Recepción son:

- PLL de Recepción
- Convertidor Serial a Paralelo
- Buffer Elástico
- Decodificador 8b/10b
- De-scrambler
- Byte un-striping
- Circuito de remoción de caracteres de control
- Buffer de recepción de paquetes de datos

Conforme se reciben los *streams* de bits, el PLL se sincroniza con la frecuencia de reloj con las que el paquete de datos fue transmitido.

Los 10b del símbolo son introducidos al Buffer elástico usando el reloj que se recuperó con el PLL. También el Buffer elástico ajusta las diferencias de frecuencia que existen entre el reloj recuperado por el PLL y el reloj original con el que se transmitió la información.

Los 10b del símbolo son convertidos al carácter de 8b en el decodificador 8b/10b. Los bits de *Start* y *Stop* son eliminados. Este dispositivo también busca errores en los datos recibidos.

El *De-Scrambler* reproduce el dato original digital aplicando el algoritmo inverso original del *scrambling*.

Los bytes de cada canal son *un-striped* para formar el byte serial que es cargado dentro del Buffer de recepción para alimentar el *Data Link Layer* [4].

### 2.2.7 Inicialización de la Aplicación e Información del Reset

PCI Express describe dos mecanismos de generación de *Reset*, el primero es el generado por el sistema mismo al cual se le llama *Reset fundamental*. El segundo mecanismo es un *In-band Reset* (comunicado vía el Link de un dispositivo al otro) al que se le llama *Hot Reset*.

El *Reset fundamental* causa que el dispositivo inicialice a las máquinas de estados del dispositivo, Hardware, puertos y registros de configuración.

Existen dos tipos de *Reset fundamental*:

*Cold Reset*: Este es un reset provocado por los voltajes de alimentación.

*Warm Reset*: Es disparado por el Hardware sin que el voltaje de alimentación sea desactivado. El diseñador decide el modo de funcionamiento de este mecanismo [4].

Existe una serie de lineamiento que se deben de cumplir cuando el *Reset Fundamental* es aplicado que se enumeran a continuación:

- La terminación del receptor deben de cumplir con los requerimientos de  $Z_{rx-high-imp-dc}$  de mínimo 200k $\Omega$ .
- Las terminaciones del transmisor deben de cumplir con los requerimientos de impedancia  $Z_{tx-dc}$  de mínimo 40 $\Omega$ , pero también puede poner a los drivers en alta impedancia.
- El transmisor mantiene un voltaje de DC en modo común entre 0 y 3.6 V
- Después de que se aplica el *Reset fundamental*:
- El receptor debe de re habilitar sus terminaciones de receptor  $Z_{rx-diff-dc}$  de 100 $\Omega$  después de 5 ms de terminado el *Reset fundamental*.
- La máquina de estados *Link Training* debe de entrar en estado *Detect* y el transmisor está listo para detectar la presencia de un receptor al otro lado de la línea.
- El transmisor mantiene un voltaje de DC en modo común entre 0 y 3.6 V.

El *Hot Reset* es propagado en la banda vía la transmisión de *TS1 Ordered-Sets* en el bit 0 del símbolo 5 en el *TS1 Ordered-Set asserted*. El *Ts1 Ordered-Set* es transmitido en todas los *Lanes* con los símbolos Link# y Lane# correctos. Estos TS1 son transmitidos de manera continua por 2 ms. Tanto el receptor como el transmisor del *Hot Reset* terminan en el estado *Detect*. *Hot Reset* en general es un *reset* generado por software [4].

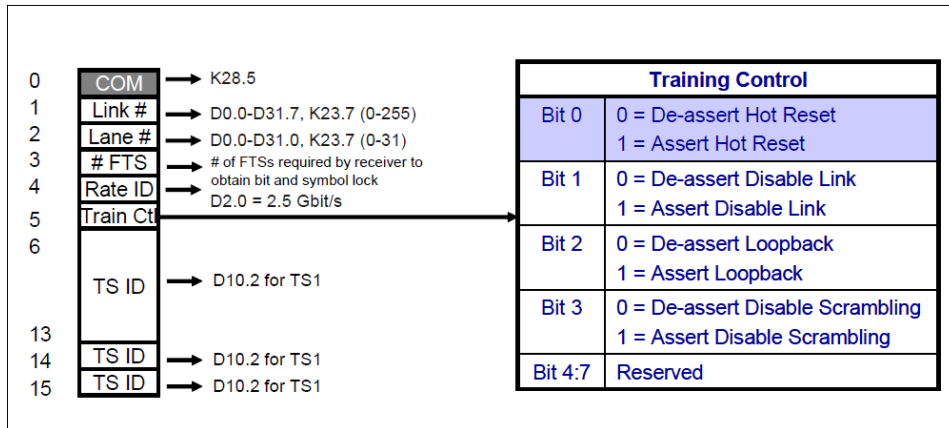


Figura 2-7: "PCI Express System Architecture" [4]

El *Hot Reset* es propagado de manera descendente (*downstream*). No se propaga de manera ascendente (*upstream*). Esto significa que solamente el *Root Complex* y el *Switch* están autorizados para generar un *Hot Reset*. Los puntos terminales no pueden generar un *Hot Reset*. Si un *Switch* recibe un *Hot Reset*, deberá resetear a todos los dispositivos que estén conectados a él además de resetearse a sí mismo [4].

### 2.2.8 2.5 GT/s Compliance Pattern



Figura 2-8: 2.5GT/s compliance pattern [4]

Las especificaciones PCI Express (V1.1 y posteriores) requiere que se capturen y procesen un millón *unit intervals* de datos para poder hacer una evaluación valida.

La compañía PCI-SIG maneja tarjetas *Compliance base* y *Compliance load* que ayudan a completar esta tarea. Estas tarjetas proporcionan una plataforma consistente para hacer mediciones eléctricas. Un set up típico es como sigue [4]:

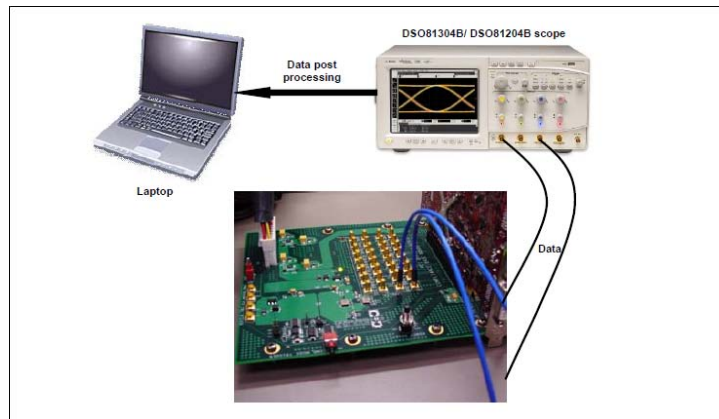


Figura 2-9: Hardware de prueba de compliance [4]

Con este tipo de *set up*, la información es capturada en el osciloscopio. Procesos posteriores son utilizados en el osciloscopio para hacer mediciones de *jitter* en el reloj de referencia y para medir el *jitter* determinístico y aleatorio en las líneas de datos. En las pruebas eléctricas, es necesario medir de manera individual cada una de las líneas, ya que cada una puede tener características diferentes. Un ejemplo del diagrama de ojo es como sigue [4]:

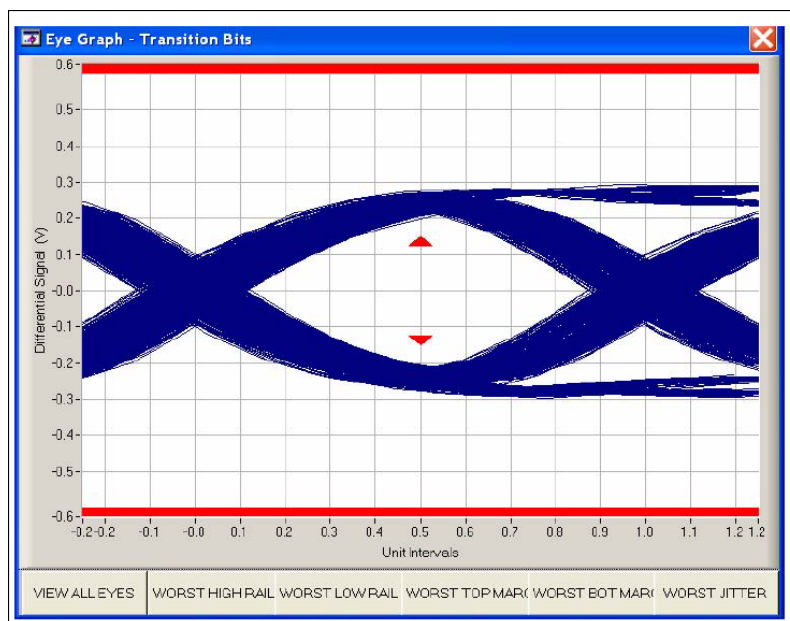


Figura 2-10: Diagrama de ojo durante la prueba de medición de jitter [4]

El análisis del diagrama de ojo es útil para medir las tolerancias de voltaje y el *jitter* [4].

### 2.3 Objetivo del Proyecto

La etapa de de-serialización tiene como objetivo el convertir datos seriales a paralelos y alinear los datos recibidos con un reloj del sistema. El sistema deberá cumplir con las especificaciones de velocidad y proponer un esquema de recuperación de reloj y datos basado en el muestreo de los datos.



La etapa de serialización tiene como objetivo convertir un bus de datos paralelos en datos en formato serie. El sistema debe cumplir con las especificaciones de velocidad y proveer una señal de reloj de transmisión para sincronizar la transmisión de datos paralelos.

La etapa analógica de recepción tiene como objetivo el compensar la atenuación en amplitud experimentada por los datos seriales al ser transmitidos a través del canal de comunicación. El sistema debe cumplir con las especificaciones de velocidad y contar con un circuito de polarización a fin de lograr la compensación a través de las diferentes esquinas de proceso, voltaje y temperatura (PVT's).

La etapa de integración del sistema tiene como objetivo crear la lógica y estructuras que permitan la comunicación de las diferentes etapas.

El sistema completo debe contar con la capacidad de poder probar los bloques digitales de manera individual y en conjunto fuera del sistema utilizando diferentes técnicas de prueba como son BIST y SCAN [4].

## 2.4 Consideraciones

1. Protocolos a ser soportados son: PEX a 2.5Gbps y SATA 1.5Gbps
2. El patrón de datos serial debe ser en código 8b10b para mantener un balance en DC.
3. La velocidad del patrón de datos es de 1.5Gbps a 2.5Gbps.
4. La frecuencia del reloj de referencia es de 0.75Ghz a 1.25Ghz para poder soportar la frecuencia de los datos.
5. La velocidad de los relojes de recepción y transmisión es 10 veces la de la frecuencia del dato.
6. Debe integrar un generador de patrón de datos interno para realizar built-in self-test (BIST).
7. Todos los módulos deben tener la opción de "bypass" con motivos de verificación. Esto es, por medio de señales de control, poder dejar pasar los datos de cada módulo sin ningún procesamiento ni adecuación al módulo siguiente.
8. El sistema debe permitir la conexión interna del dato paralelo obtenido por el de-serializer hacia el serializer para poder crear un lazo interno que permita convertir el mismo dato nuevamente a un patrón serial por el transmisor, con motivos de prueba.
9. Consideramos que **Cadence** estará disponible todos los días del año para el desarrollo del proyecto.
10. La tecnología CMOS a ser usada será de 180 nm (proceso cmrf7sf de IBM), con licencia de MOSIS.

11. El bloque del sistema SerDes que voy a desarrollar como trabajo de obtención de grado será el del Transmisor de dos canales (+Tx, -TX)

## **2.5 Restricciones**

- 1).- El proyecto deberá realizarse en poco más de medio año.
- 2).- El equipo de trabajo es de 6 personas.
- 3).- La carga de trabajo deberá de estar balanceada para que no se acumulen varios entregables en un corto plazo.
- 4).- Las personas que están desarrollando este proyecto no son estudiantes de tiempo parcial por lo que el tiempo de trabajo en el desarrollo del mismo es limitado.

Este proceso de diseño de un sistema SerDes, es un proceso retador por el número de integrantes del proyecto y por el tiempo que se tiene para el mismo, esperamos poder llevar a buen término el proceso de diseño y layout, y tener tiempo suficiente para llegar a la implementación.



### 3 Diseño esquemático del módulo Transmisor de datos del sistema SerDes

El transmisor de datos que se diseñara debe recibir señales de datos provenientes de alguna fuente interna o externa, que representa la información que se desea enviar al receptor, y entregar ella misma la trama de datos hacia fuera del chip. El circuito debe tener una impedancia, amplitud y forma programable (ecualización). Estas características se logran con la técnica de terminación segmentada de resistencias en serie, multiplexores, buffers, circuitos amarradores de nivel, una celda de retardo y lógica de control. El pre énfasis requiere de una copia de la señal de entrada invertida y retrasada en el tiempo.

El diagrama a bloques del circuito descrito en el párrafo anterior se muestra en la Figura 3-1:

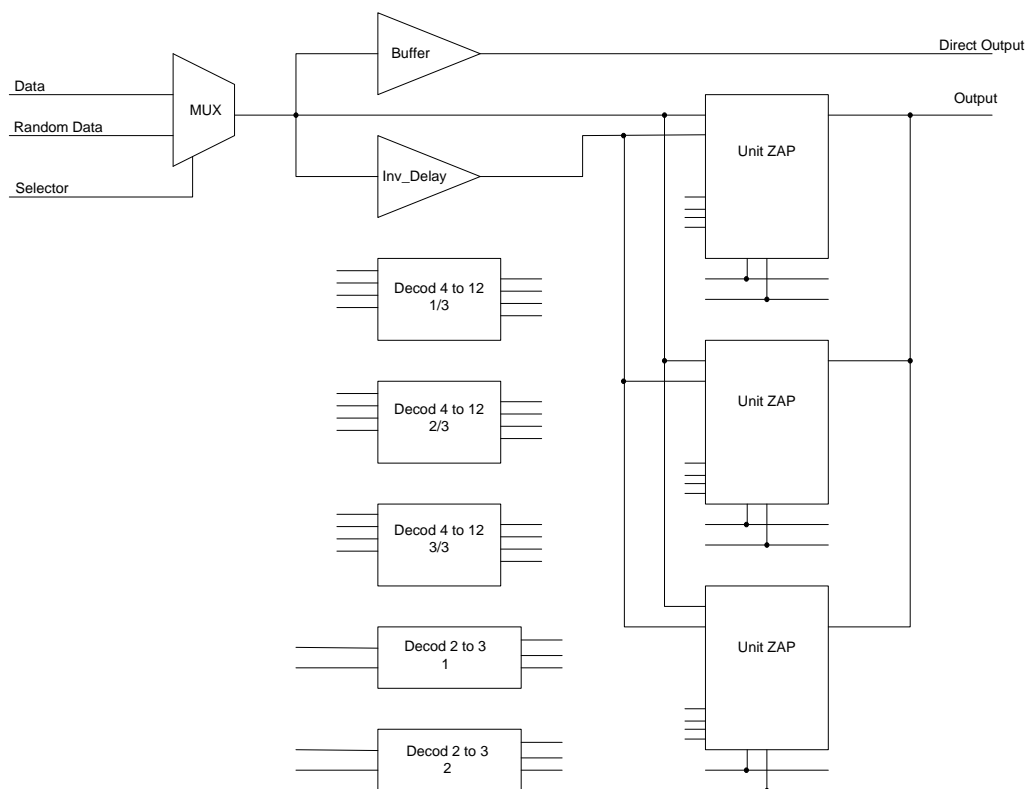


Figura 3-1: Diagrama a bloques del circuito transmisor de datos [5]

Consta de las siguientes entradas; Data, Random Data, Selector, 4 entradas de control para la impedancia de acople con la línea de transmisión, 2 entradas de control para la amplitud y 2 entradas de control para el pre-énfasis.

Consta de las siguientes salidas; Direct\_Output y Output

Como puede observarse de la Figura 3-1, el circuito transmisor está compuesto por un multiplexor 2 a 1, un buffer de retardo-inversor, un buffer de salida directa, un decodificador de 4 a 12, un decodificador de 2 a 6 y tres unidades de control de amplitud, impedancia y pre-énfasis, llamadas ZAP.

### 3.1 Desarrollo del Módulo transmisor

Todos los bloques del circuito transmisor se construyen a partir de bloques básicos repetitivos, por esta razón, el proceso de diseño del circuito transmisor emplearemos una metodología *Bottom – Up*.

La gran ventaja de esta topología SSSTT se debe a su bajo consumo de operación y su habilidad de manejar un rango grande de terminaciones de voltaje, que los hace candidatos ideales para transmisores de múltiples estándares, se basan enteramente en dispositivos digitales que son optimizados para operar en altas velocidades y su escalamiento sigue desarrollándose con el desarrollo de la tecnología [6][7].

Primero se presenta el diseño de los circuitos básicos que componen los diversos sub-módulos del circuito transmisor de datos y enseguida, para efectos de validación del diseño, se presentan resultados de simulación a nivel esquemático (pre-layout) de cada uno de los sub-módulos.

En general para el diseño de circuitos digitales se toman en cuenta las siguientes consideraciones; Se obtiene una característica de transferencia simétrica cuando los dispositivos se diseñan para que tengan parámetros de transconductancia iguales, una condición llamada *coincidencia*. Como la  $\mu_n$  es dos a cuatro veces la  $\mu_p$ , la coincidencia se logra haciendo  $(W/L)_p$  dos a cuatro veces la  $(W/L)_n$ .

Normalmente los dos dispositivos tienen la misma longitud de canal  $L$ , la cual se establece al mínimo admisible para la tecnología del proceso dada. El ancho mínimo del transistor NMOS es de una y media a dos veces  $L$ , y el ancho del transistor PMOS de dos a tres veces  $L$ .

#### 3.1.1 Diseño del circuito Inversor

La Figura 3-2 presenta el esquemático a nivel transistor del circuito inversor. Para la asignación de los tamaños de los transistores se consideró lo siguiente: tomando en cuenta que en el proceso IBM\_cmrf7sf la  $k_{p_n} = 157.8 \mu A/V^2$  y la  $k_{p_p} = 33.2 \mu A/V^2$ , la relación de los tamaños de los transistores  $n$  y  $p$  debe ser  $(W/L)_n = (k_{p_n}/k_{p_p})(W/L)_p$ ; de esta manera, si usamos una  $L = L_{\min} = 180\text{nm}$ , la  $W_n = 1.2 \mu\text{m}$  y la  $W_p = 3.0 \mu\text{m}$  [8].

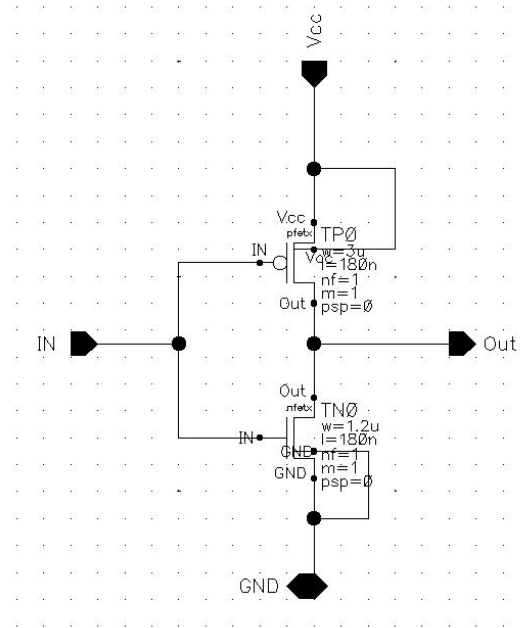


Figura 3-2: Esquemático del circuito inversor [5]

### 3.1.2 Diseño de compuerta NOR de dos entradas (NOR2)

Las Figura 3-3 presenta el esquemático a nivel transistor de la compuerta NOR de 2 entradas y la Tabla 3-1 presenta las dimensiones de los transistores de la misma:

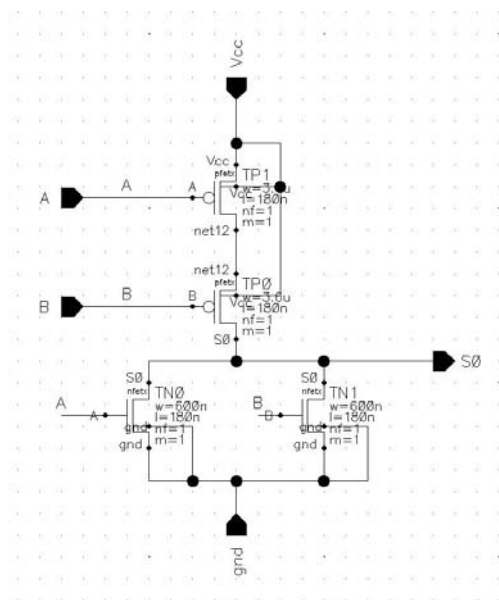


Figura 3-3: Esquemático del circuito de compuerta NOR2 [5]

Transistor	W	L
TP0	3.6um	180nm
TP1	3.6um	180nm
TN0	600nm	180nm
TN1	600nm	180nm

Tabla 3-1: Dimensiones de los transistores del circuito compuerta NOR2

### 3.1.3 Diseño de NAND de dos entradas (NAND2)

La Figura 3-4 y la Tabla 3-2 presentan respectivamente el esquemático a nivel transistor y las dimensiones de la compuerta NAND de 2 entradas:

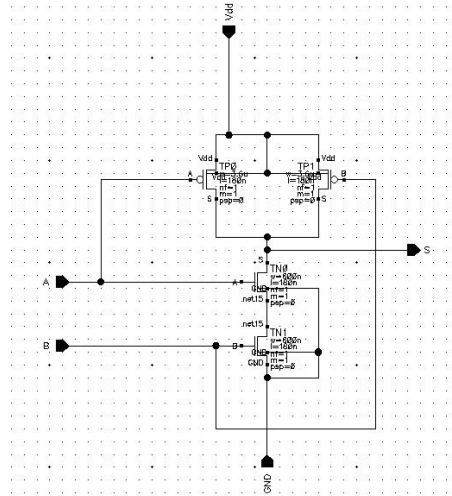


Figura 3-4: Esquemático del circuito de compuerta NAND2 [5]

Transistor	W	L
TP0	3.6um	180nm
TP1	3.6um	180nm
TN0	600nm	180nm
TN1	600nm	180nm

Tabla 3-2: Dimensiones de los transistores del circuito de compuerta NAND2

### 3.1.4 Diseño compuerta NOR de tres entradas (NOR3)

La Figura 3-5 y la **Error! Reference source not found.** presentan respectivamente el esquemático a nivel transistor y las dimensiones de los transistores de la compuerta NOR de 3 entradas:

Transistor	W	L
TP0	3.6um	180nm
TP1	3.6um	180nm
TP2	3.6um	180nm
TN0	600nm	180nm
TN1	600nm	180nm
TN3	600nm	180nm

Tabla 3-3: Dimensiones de los transistores del circuito de compuerta NOR3

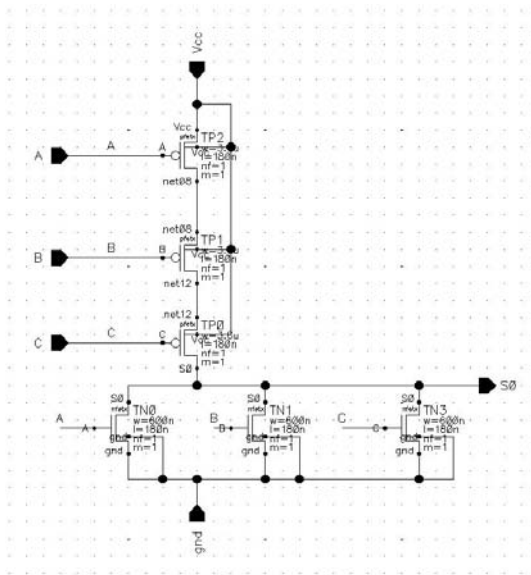


Figura 3-5: Esquemático del circuito de compuerta NOR3 [5]

### 3.1.5 Diseño compuerta NAND de tres entradas (NAND3)

La Figura 3-6 y la Tabla 3-4 presentan respectivamente el esquemático a nivel transistor y las dimensiones de los transistores de la compuerta NAND de 3 entradas:

Transistor	W	L
TP0	3.6um	180nm
TP1	3.6um	180nm
TP3	3.6um	180nm
TN0	600nm	180nm
TN1	600nm	180nm
TN4	600nm	180nm

Tabla 3-4: Dimensiones de los transistores del circuito de compuerta NAND3



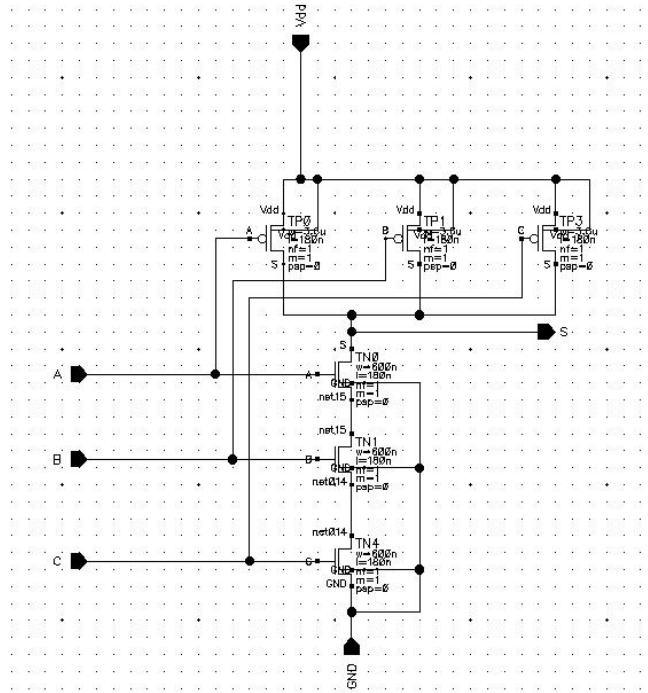


Figura 3-6: Esquemático del circuito de compuerta NAND3 [5]

### 3.1.6 Diseño del Buffer de tres estados

La Figura 3-7 presenta el esquemático a nivel transistor del circuito del buffer de tres estados, éste buffer se usará para implementar el bloque de la celda básica de la unidad de control de impedancia. La Tabla 3-5 muestra las dimensiones del mismo:

Transistor	W	L
TP0	8.1um	180nm
TP1	8.1um	180nm
TP2	20.4um	180nm
TN0	9um	180nm
TN1	9um	180nm
TN2	23.8um	180nm
TN3	23.8um	180nm

Tabla 3-5: Dimensiones de los transistores del circuito Buffer de tres estados

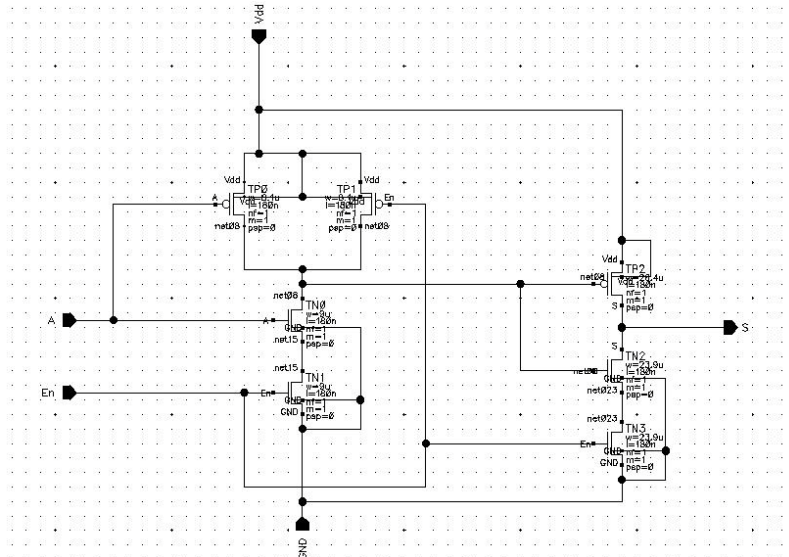


Figura 3-7: Esquemático del circuito Buffer de tres estados [5]

Para esta celda, los transistores PMOS y NMOS tienen la mínima distancia permitida en la tecnología para el largo, y los anchos fueron ajustados para cumplir con la aportación de la resistencia del 10% de la impedancia total de la salida en el valor central de la calibración (es posible mediante la Ecuación 3-1, hacer una estimación de la resistencia aportada por los transistores de salida del buffer de tercer estado al valor de la impedancia total del circuito transmisor). La compuerta NAND es una compuerta simple de dos entradas y con dimensiones mínimas en los largos de los transistores; las  $W$ , o los anchos, son de 8.1  $\mu\text{m}$  y 9  $\mu\text{m}$  para los transistores tipo P y tipo N, respectivamente. La razón de estas dimensiones para la compuerta NAND es que esta debe ser capaz de excitar a los transistores de la etapa de salida del buffer de tercer estado, y debido a que estos son de dimensiones grandes pues representan una gran carga para la compuerta.

$$\text{Ecuación 3-1 } R_{eq} = \frac{3}{4} \frac{V_{DD}}{I_{DSAT}} \left( 1 - \frac{7}{9} \lambda V_{DD} \right)$$

### 3.1.7 Diseño del Inversor-Retardador

La Figura 3-8 presenta el esquemático a nivel transistor del circuito retardador. Para la realización de los cambios de pre-énfasis de las señales transmitidas, es necesario implementar una celda de retardo inversora. Esta se implementó con tres inversores en cascada, los cuales se ajustaron para obtener un retardo del 10% del periodo de la señal que se está transmitiendo, es decir, un retardo de 1ns; con esta celda de retardo se genera la señal de *Weak Data*, la cual se utiliza por las unidades ZAP para generar los cambios de pre-énfasis. Las dimensiones de las celdas de retardo fueron

aproximadas mediante la Ecuación 3-1 y las ecuaciones de los tiempos de propagación expuestas en [10], sin embargo el valor final de las dimensiones de los transistores se obtuvo mediante pruebas de simulación. La Tabla 3-6 muestra estas dimensiones:

Transistor	W	L
TP0	680nm	360nm
TP1	3.6um	360nm
TP2	3.9um	180nm
TN0	360nm	360nm
TN1	1.62um	360nm
TN2	1.8um	180nm

Tabla 3-6: Dimensiones de los transistores del circuito Retardador Inversor

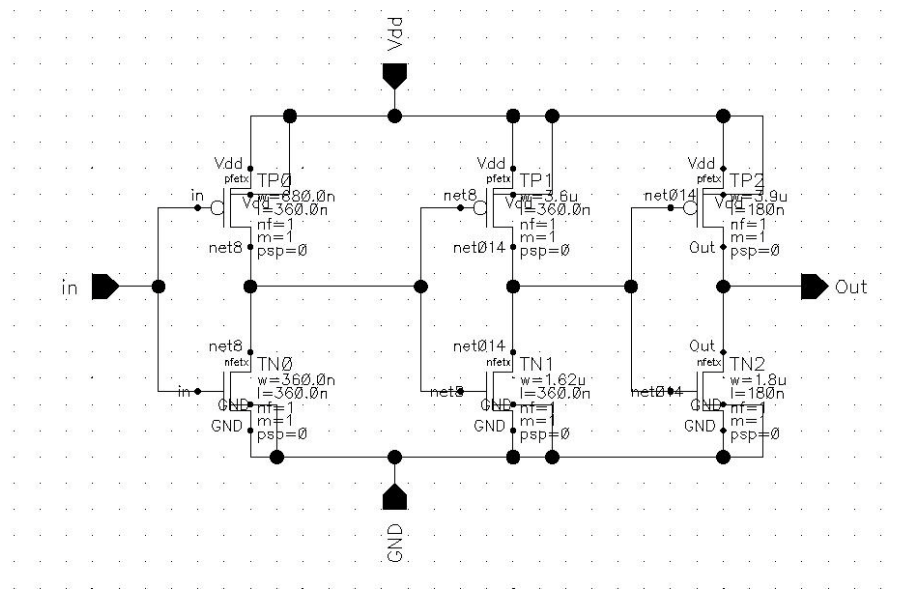


Figura 3-8: Esquemático del circuito Retardador-Inversor [5]

### 3.1.8 Diseño del decodificador de 4 entradas y 12 salidas, (DECO 4 a 12)

La Figura 3-9 muestra el diseño a bloques del decodificador de 4 entradas y 21 salidas, las 4 entradas son comunes para los tres bloques y las doce salidas se forman al sumar las 4 salidas de cada uno de ellos:

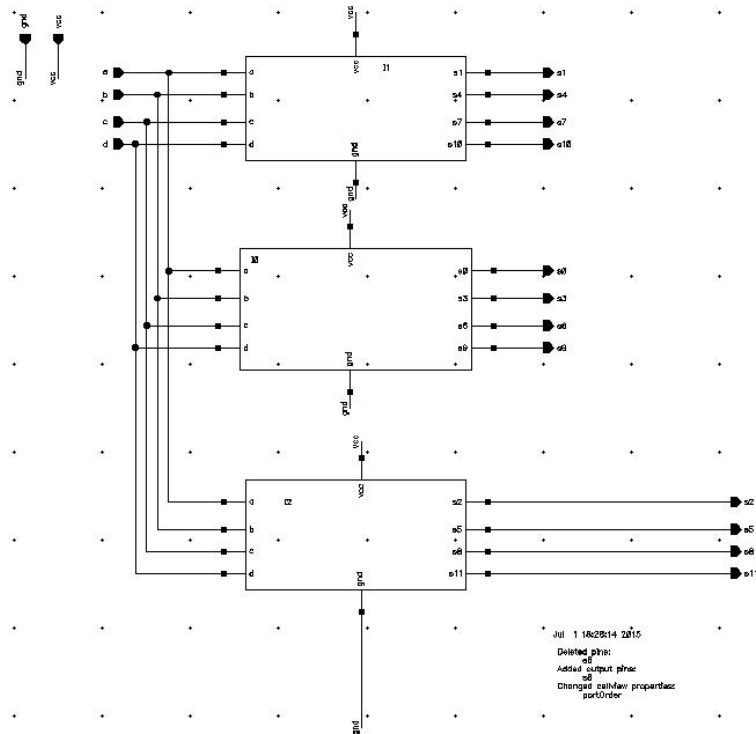


Figura 3-9: Tres bloques integran el decodificador de 4 a 12 [5]

Como se indica en la Figura 3-10, el diseño interno del bloque 1 está constituido por compuertas NAND2, NAND3, NOR2, NOR 3 y compuertas inversoras.

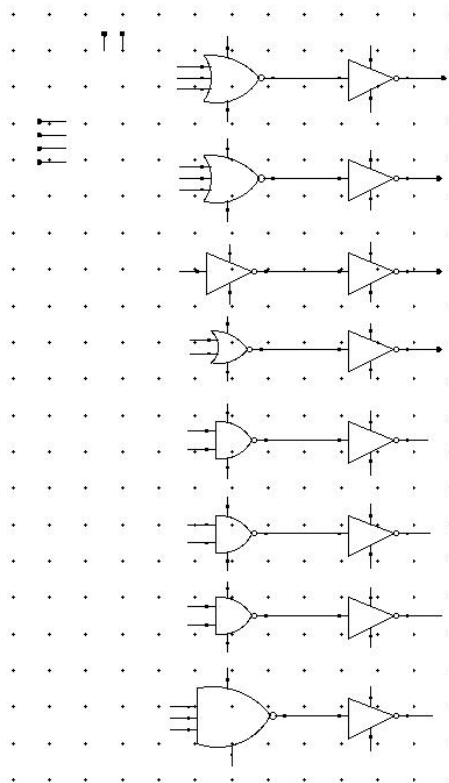


Figura 3-10: Diseño interno del bloque 1 [5]

Como se indica en la Figura 3-11, el diseño interno del bloque 2 está constituido por compuertas NAND2, NAND3, NOR2 y compuertas inversoras.

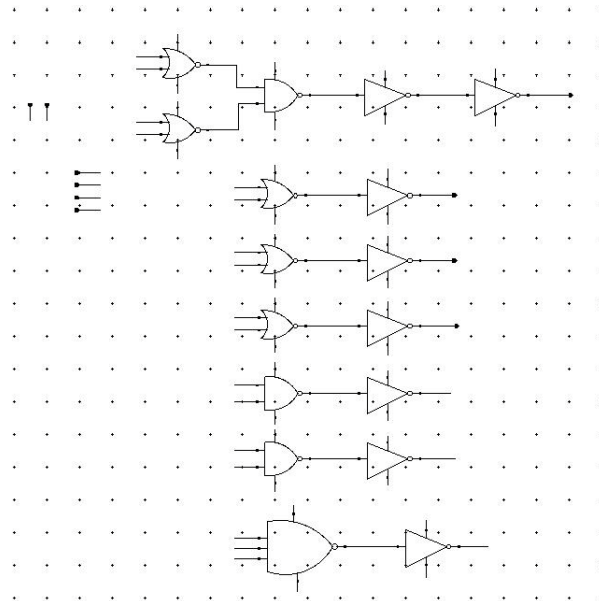


Figura 3-11: Diseño interno del bloque 2 [5]

Como se indica en la Figura 3-12, el diseño interno del bloque 3 está constituido por compuertas NAND2, NOR3, NOR2 y compuertas inversoras.

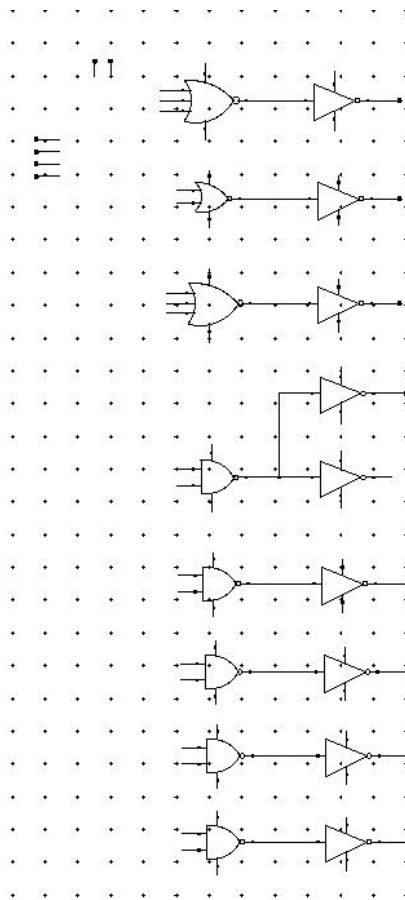


Figura 3-12: Diseño interno del bloque 3 [5]

### 3.1.9 Diseño del decodificador de 2 entradas y 3 salidas, (DECO 2 a 3)

La Figura 3-13 presenta el esquemático a nivel compuerta del DECO 2 a 3, este consiste de compuertas NOR2, NAND2 e inversores

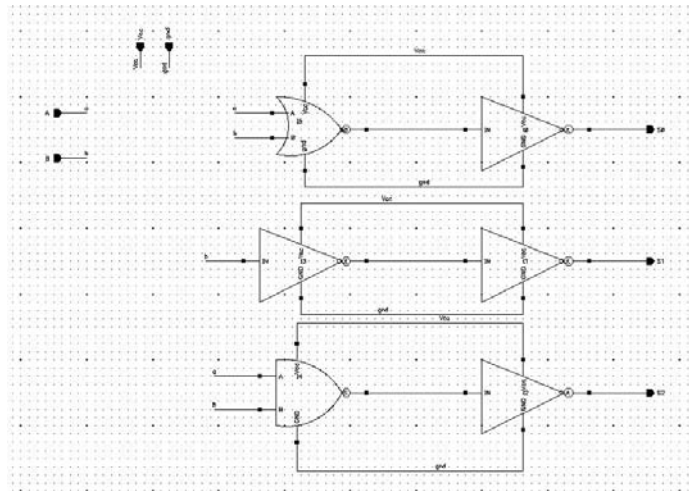


Figura 3-13: Decodificador 2 entradas 3 salidas [5]

### 3.1.10 Diseño del Multiplexor (MUX 2 a 1)

La Figura 3-14 presenta el esquemático a nivel transistor del MUX 2 a 1 y la Tabla 3-7 las dimensiones de los transistores que lo componen:

Transistor	W	L
TP0	1.62um	180nm
TP1	1.62um	180nm
TP2	1.62um	180nm
TP5	1.62um	180nm
TP7	540nm	180nm
TP8	540nm	180nm
TN0	900nm	180nm
TN1	900nm	180nm
TN2	900nm	180nm
TN5	900nm	180nm
TN6	300nm	180nm
TN7	300nm	180nm

Tabla 3-7: Dimensiones de los transistores del Circuito MUX 2 a 1

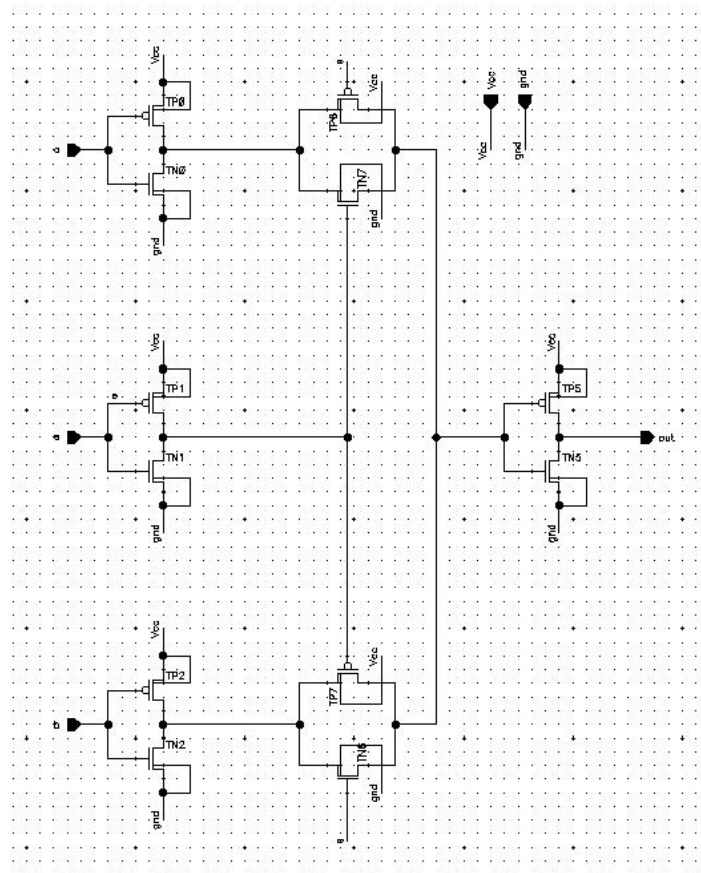


Figura 3-14: Esquemático del circuito Mux 2 a 1 [5]

Una primera aproximación en el dimensionamiento de los transistores se puede encontrar mediante la Ecuación 1. A partir de esta aproximación, es necesario realizar una serie de ajustes en las dimensiones de los transistores para poder lograr la respuesta deseada en cuanto a tiempos de subida y tiempos de bajada.

### 3.1.11 Diseño del circuito amarrador (tied)

La Figura 3-15 presenta el esquemático a nivel transistor del circuito amarrador a valor Alto o valor Bajo ya la Tabla 3-8 las dimensiones de los transistores que lo componen:

Transistor	W	L
TP0	1.08um	180nm
TP1	1.08um	180nm
TP2	1.08um	180nm
TN0	600nm	180nm
TN1	600nm	180nm
TN2	600nm	180nm

Tabla 3-8: Dimensiones de los transistores del Circuito Tied

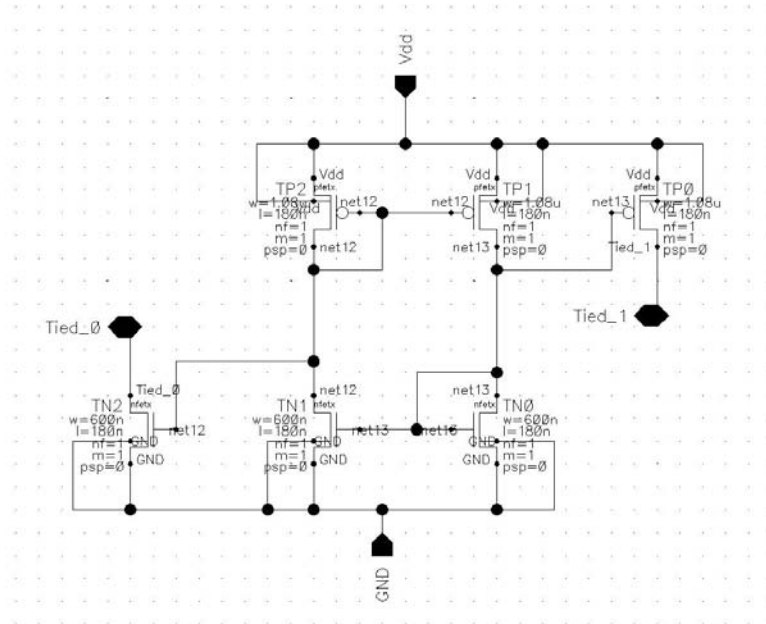


Figura 3-15: esquemático del circuito Amarrador (Tied) [5]

### 3.1.12 Diseño de la unidad de control ZAP

Las unidades ZAP son los bloques que controlan todas las características del dispositivo, es decir su amplitud, impedancia y pre-énfasis. Este sub-módulo se compone de cuatro circuitos MUX 2 a 1, un circuito buffer de entrada, cuatro unidades básicas, cuatro buffer de control de las unidades básicas y un circuito amarrador. La Figura 3-16 presenta el diagrama interno de la unidad ZAP:

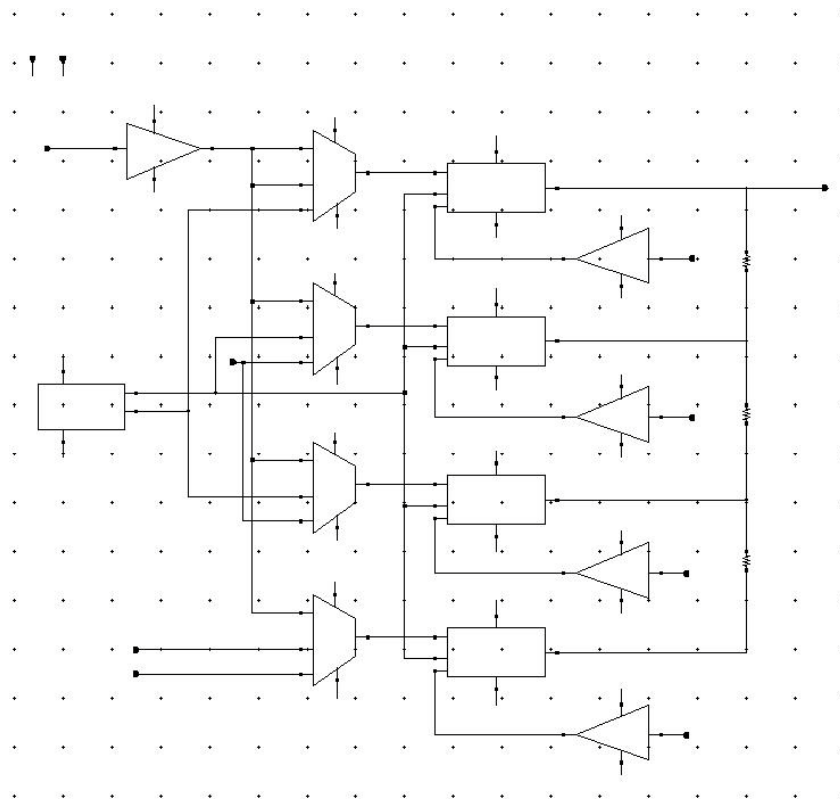


Figura 3-16: Diagrama a bloques de la unidad ZAP [5]



Cada unidad ZAP está compuesta de 4 celdas básicas formadas por dos buffers de tres estados y dos resistores discretos (Figura 3-17):

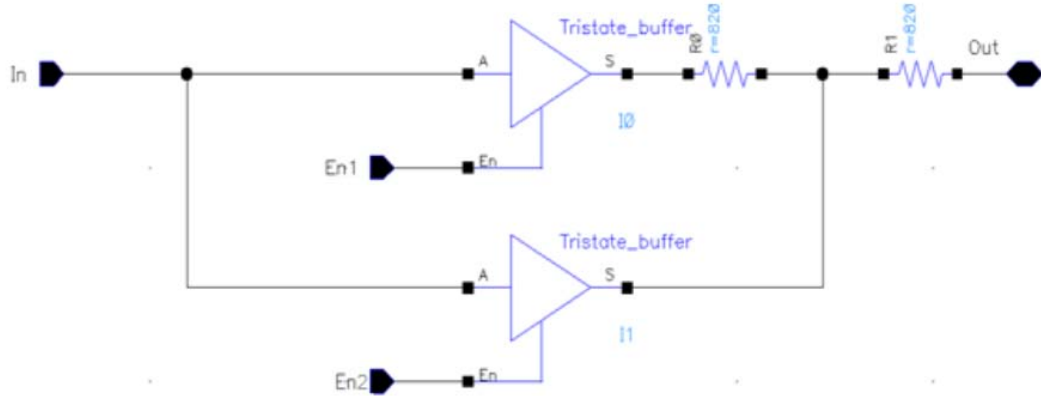


Figura 3-17: Circuito de celda básica de la unidad ZAP [5].

El diagrama interno de los buffers de tres estados así como las dimensiones de los transistores de ese circuito se presentaron en la Figura 3-7 y Tabla 3-5 respectivamente.

### 3.1.13 Circuito Transmisor de datos

Finalmente, integramos todos los sub-módulos del circuito transmisor en un solo bloque, la Figura 3-18 muestra este bloque con sus alimentaciones y señales requeridas para verificar su respuesta:

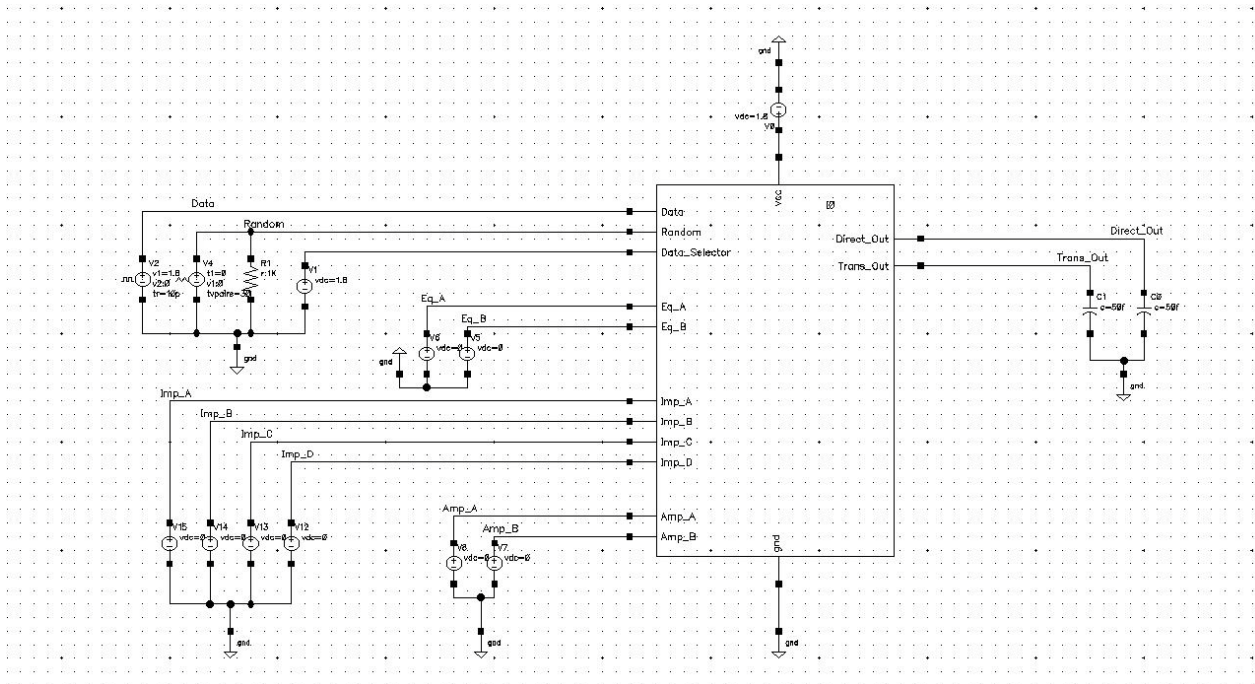


Figura 3-18: Esquemático test bench del módulo transmisor de datos [5]





La configuración para las señales Imp\_A, Imp\_B, Imp\_C e Imp\_D se muestra en la Tabla 4-2:

Signal	Voltage		Period	Rise Time	Fall Time	Pulse Width
	1	2				
Imp_A	1.8	0	5.32ns	10ps	10ps	2.66ns
Imp_B	1.8	0	10.66ns	10ps	10ps	5.32ns
Imp_C	1.8	0	21.2ns	10ps	10ps	10.66ns
Imp_D	1.8	0	42.4ns	10ps	10ps	21.2ns

Tabla 4-2: Configuración de fuentes de entrada

La frecuencia de las señales va disminuyendo en factor de 2, con el fin de obtener todas las combinaciones posibles de las cuatro señales y poder generar la tabla de verdad de datos de entrada completa, esto se muestra numéricamente en la Tabla 4-3 y a nivel de señal en la Figura 4-1:

Imp_D	Imp_C	Imp_B	Imp_A
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Tabla 4-3: Combinación digital numérica de entradas

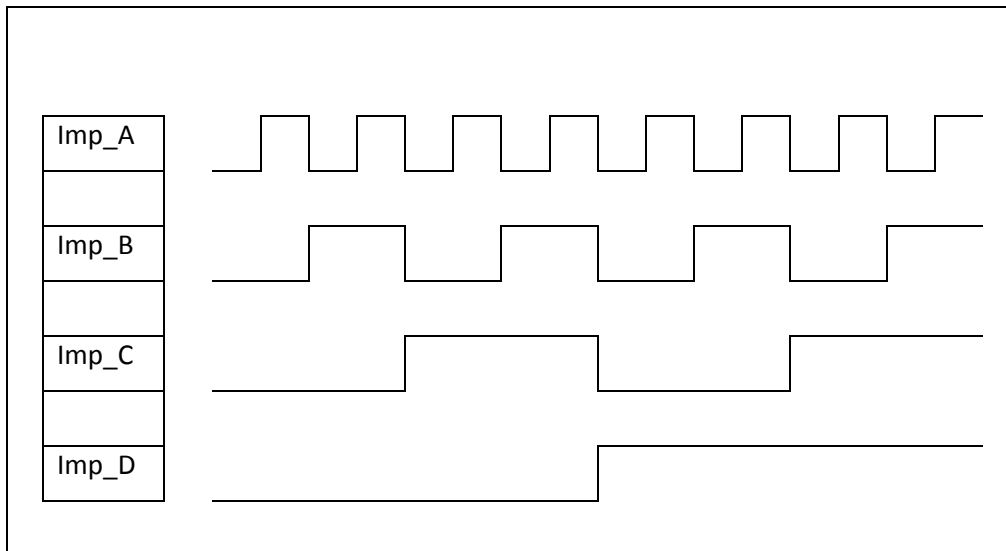


Figura 4-1: Combinación digital a nivel señal de entradas

Haciendo referencia a la Figura 3-18, la configuración de los pines de entrada Data, Random y Data Selector corresponden a la señal se quiere utilizar como entrada de datos para el transmisor, la señal Data tiene conectada una fuente que envía un patrón de datos a una frecuencia constante, esta frecuencia debe de ser la del estándar del protocolo de comunicación de datos (PCIe1 en este caso), para nuestro caso corresponde a una frecuencia de 1.5GHz, la configuración de la fuente se muestra a continuación en la Tabla 4-4:

Signal	Voltage 1	Voltage 2	Period	Rise Time	Fall Time	Pulse Width
Data	1.8	0	1.33ns	10ps	10ps	0.667ns

Tabla 4-4: Configuración señal Data

La fuente Random de la Figura 3-18 tiene un elemento al cual se la ha programado una señal de datos aleatoria, la explicación de cómo programar la fuente de datos aleatoria se dará en la sección de la prueba respectiva, la señal Data selector se encarga de seleccionar cuál de estas dos señales entran al Transmisor. Con un valor de 0V en la fuente V1 de la Figura 3-18, la selección de entrada es para Data, mientras que con un valor de 1.8V en la fuente V1 de la Figura 3-18, la selección de entrada es para la señal Random.

Para poder medir el impacto en la modificación de impedancia al correr la simulación, es conveniente cambiar el capacitor que se tiene como carga a la salida *Trans\_Out* del transmisor por una resistencia (ver Figura 3-18), de esta manera, observando el cambio en la amplitud de la señal podemos establecer que impedancias se están dando en la línea de transmisión. La impedancia de la línea debe estar en su valor central en 100  $\Omega$ , por lo que, si ponemos una resistencia de 100  $\Omega$  como carga,

estaremos obteniendo valores de voltaje de alrededor de un divisor de voltaje entre dos. En las siguientes secciones se explica de manera numérica y grafica este efecto. Para propósitos ilustrativos, revisaremos tres casos de nivel de impedancia generados por el circuito.

#### 4.1.1 Pruebas del control impedancia, condición de alta impedancia:

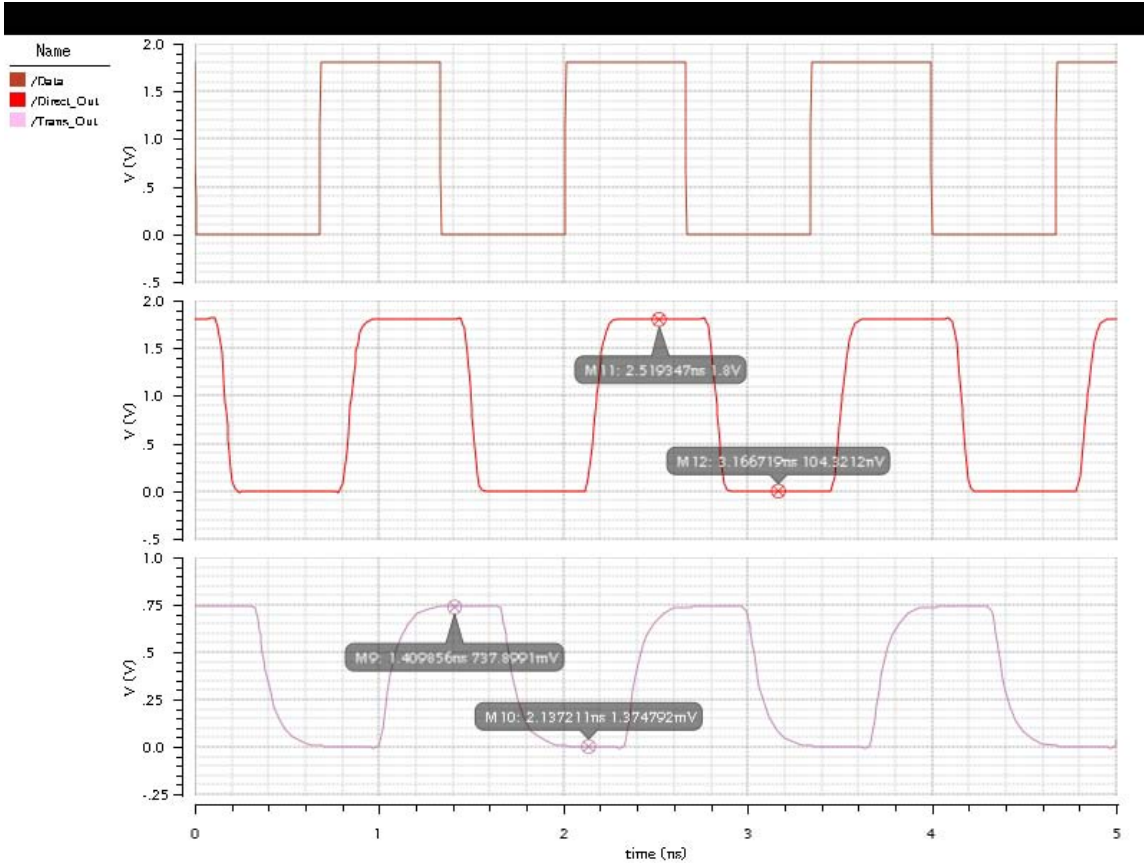


Figura 4-2: Verificación de cambio de impedancia (Impedancia Alta)

La tercera curva de la Figura 4-2 en color rosa que corresponde a la salida del transmisor; la amplitud del voltaje de salida es  $V_{out} = 737.899 \text{ mV}$ , utilizando la Ecuación 4-1 para el divisor de voltaje con la resistencia  $R_B$  despejada y sabiendo que tenemos un  $V_{in} = 1.8V$  y una  $R_A = 100 \Omega$ , calculamos una resistencia resultante  $R_B = 144\Omega$  que es la impedancia que el transmisor tiene configurada en acople con la línea de transmisión. Vemos que se tiene configurada una impedancia alta respecto de los  $100 \Omega$  nominales de la línea de transmisión.

$$\text{Ecuación 4-1} \quad R_B = \left[ \frac{V_{in}}{V_{out}} * R_A \right] - R_A$$

#### 4.1.2 Pruebas del control impedancia, condición de impedancia intermedia:

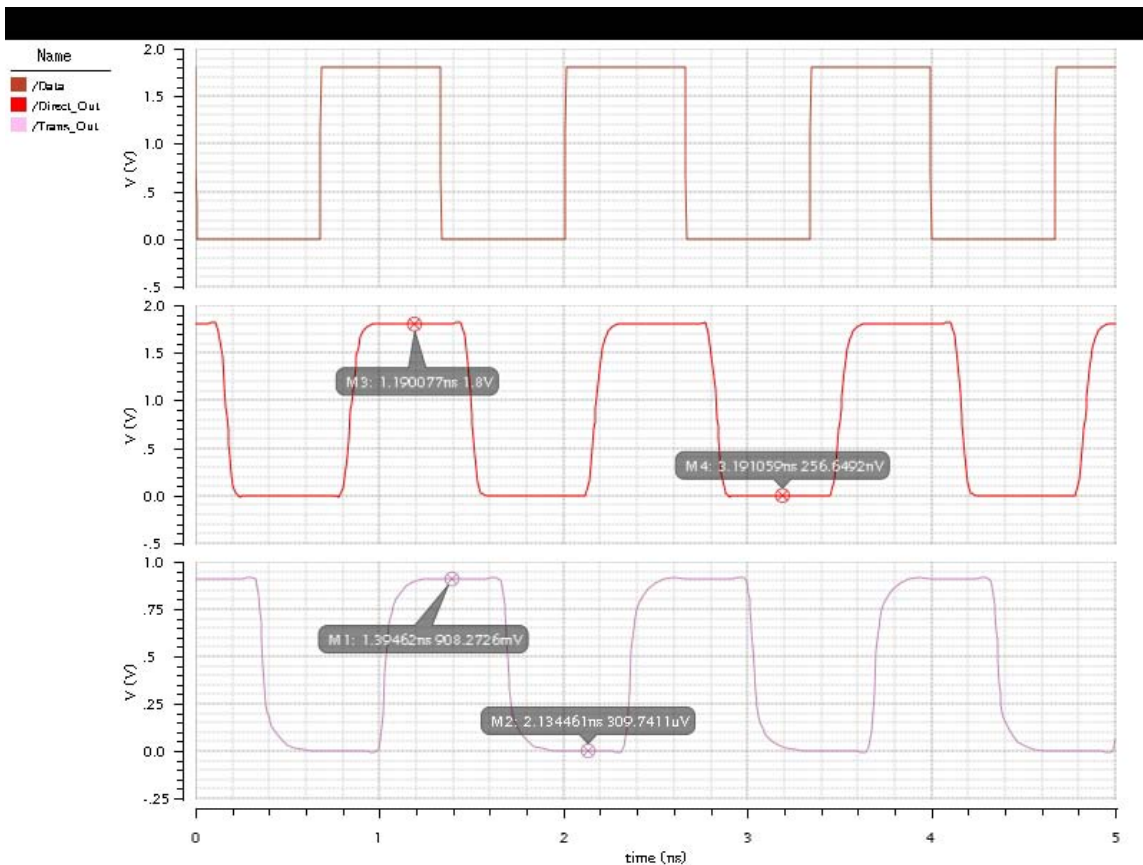


Figura 4-3: Verificación de cambio de impedancia (Impedancia Intermedia)

Repitiendo el proceso descrito en la sección anterior pero esta vez para un voltaje de salida de  $V_{out} = 908.272 \text{ mV}$  que se observa en la Figura 4-3 (curva en color rosa) obtenemos una impedancia  $R_B = 98.1 \Omega$ . Esta es una impedancia intermedia muy cercana a los  $100 \Omega$  nominales de la línea de transmisión.

### 4.1.3 Pruebas del control impedancia, condición de baja impedancia:

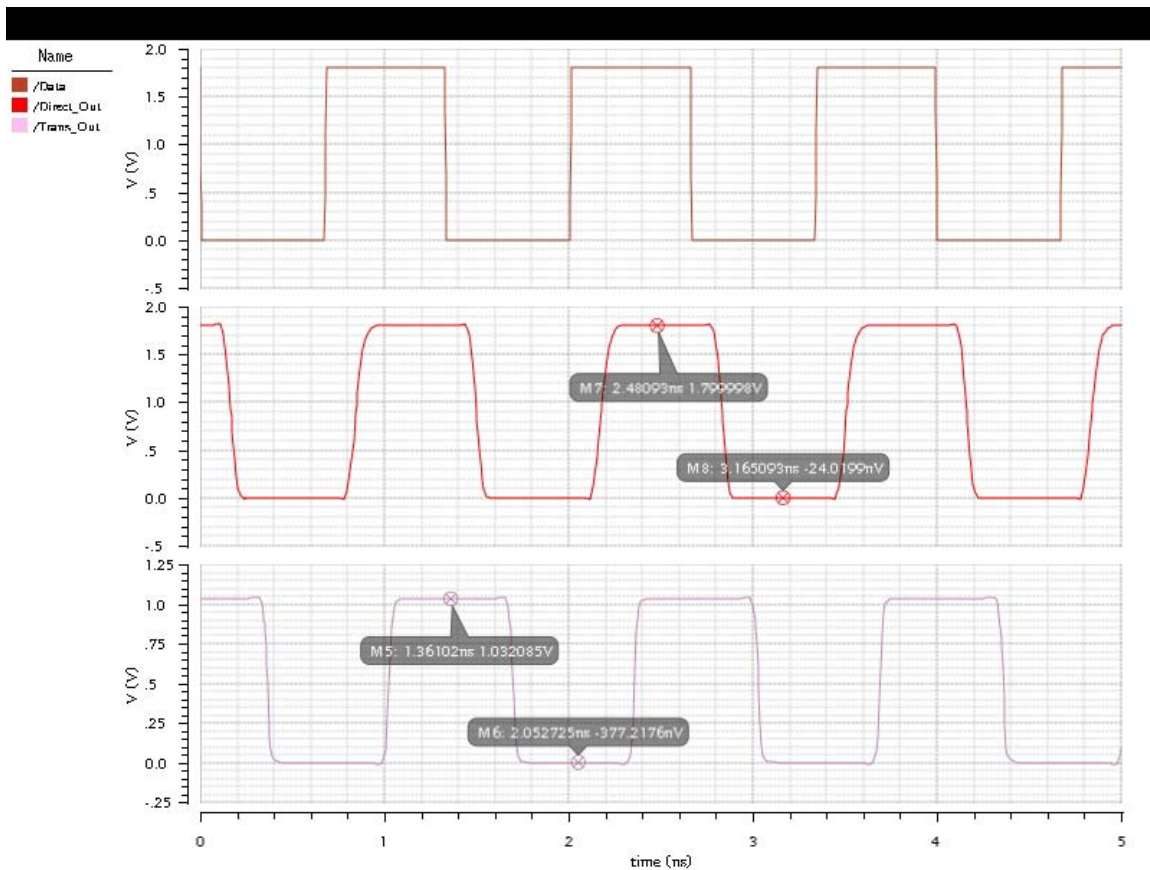


Figura 4-4: Verificación de cambio de impedancia (Impedancia Baja)

Nuevamente para un voltaje de salida de  $V_{out} = 1.032 \text{ mV}$  que se observa en la Figura 4-4 (curva en color rosa) obtenemos una impedancia igual a  $R_B = 74.4\Omega$ , que es una impedancia por debajo de los  $100 \Omega$  nominales de la línea de transmisión.

### 4.2 Setup del Control de Amplitud del Transmisor:

Nuevamente haciendo referencia a la figura Figura 3-18 para explicar la configuración de las fuentes y señales, el setup de la verificación del control de amplitud deberá de inhabilitar las variaciones de impedancia y pre-énfasis como se muestra en la Tabla 4-5:

Signal	Vdc
Imp_A	0V
Imp_B	0V
Imp_C	0V
Imp_D	0V
Eq_A	0V
Eq_B	0V

Tabla 4-5: Configuración de fuentes de Impedancia y Pre énfasis.



Con esto, la única variación que observaremos durante la simulación del transmisor se deberá solo al patrón de cambios que pondremos en las entradas Amp\_A y Amp\_B que corresponden al control de Amplitud de la línea de transmisión (Figura 3-18).

La configuración para las señales Amp\_A y Amp\_B se muestra en la Tabla 4-6:

Signal	Voltage 1	Voltage 2	Period	Rise Time	Fall Time	Pulse Width
Amp_A	1.8	0	5.32ns	10ps	10ps	2.66ns
Amp_B	1.8	0	10.66ns	10ps	10ps	5.32ns

Tabla 4-6: Configuración de fuentes de prueba Amp\_A y Amp\_B

La frecuencia de las señales van disminuyendo en factor de 2, con el fin de obtener todas las combinaciones posibles entre las dos señales para poder generar la tabla de verdad de datos de prueba, la Tabla 4-7 y la Figura 2-1 muestran las señales de prueba de manera numérica y a nivel señal:

Amp_B	Amp_A
0	0
0	1
1	0
1	1

Tabla 4-7: Combinación numérica de fuentes de prueba Amp\_A y Amp\_B

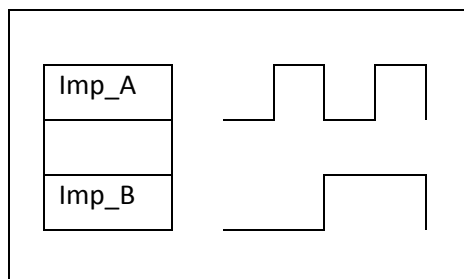


Figura 4-5: Combinación nivel señal de fuentes de prueba Amp\_A y Amp\_B

Haciendo referencia a la Figura 3-18, la configuración de los pines de entrada Data, Random y Data Selector corresponden a la señal se quiere utilizar como entrada de datos para el transmisor, hágase referencia a la tabla Tabla 4-4 para la configuración de la señal Data, ya que es la misma configuración que se usa en esa sección.

La fuente Random de la Figura 3-18 tiene un elemento al cual se la ha programado una señal de datos aleatoria, la explicación de cómo programar la fuente de datos aleatoria se dará en la sección de la prueba respectiva, la señal Data selector se encarga de seleccionar cuál de estas dos señales entran al Transmisor. Con un valor de 0V en la fuente V1 de la Figura 3-18, la selección de entrada es para Data,

mientras que con un valor de 1.8V en la fuente V1 de la Figura 3-18, la selección de entrada es para la señal Random.

#### 4.2.1 Pruebas del control de amplitud:

La Figura 4-6 Muestra el resultado de la prueba de control de amplitud de la señal que se desea transmitir:

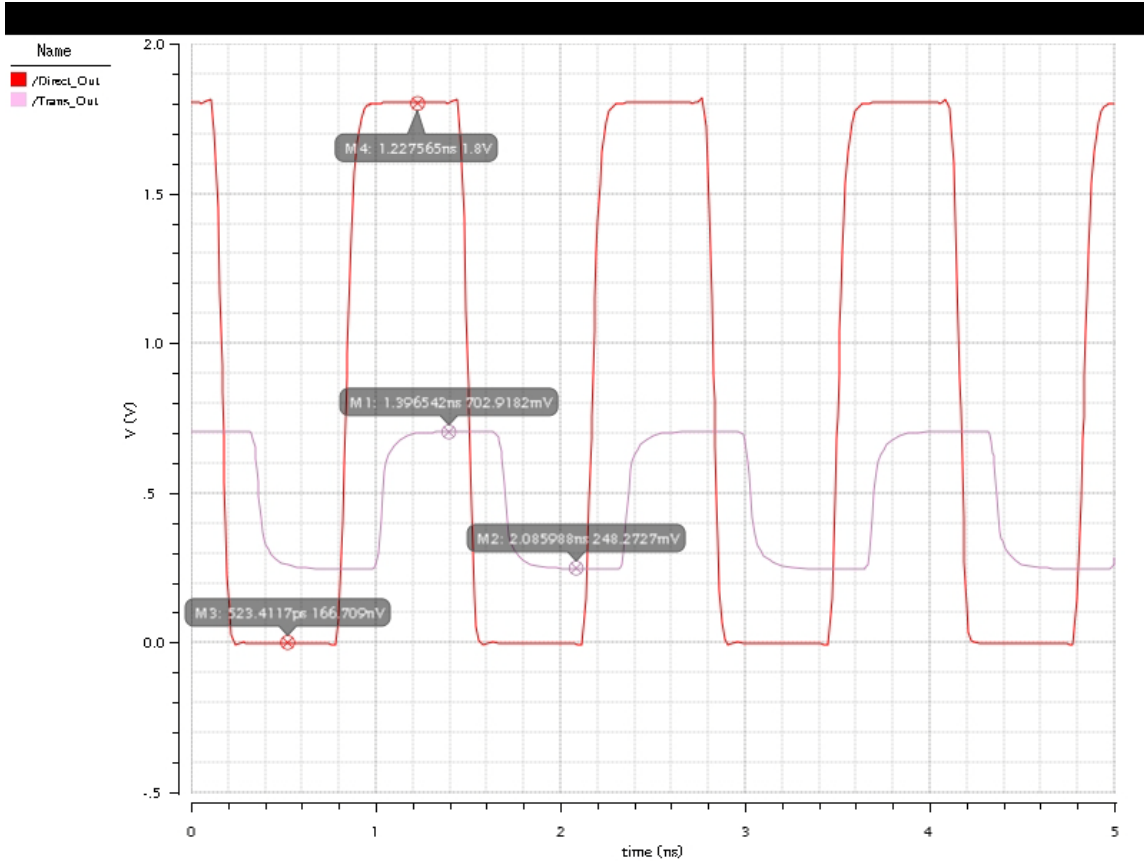


Figura 4-6: Verificación de control de amplitud

### 4.3 Setup del Control de Pre Énfasis del Transmisor:

Nuevamente haciendo referencia a la figura Figura 3-18 para explicar la configuración de las fuentes y señales, el setup de la verificación del control de pre-énfasis deberá de inhabilitar las variaciones de impedancia y amplitud como se muestra en la Tabla 4-8:

Signal	Vdc
Imp_A	0V
Imp_B	0V
Imp_C	0V
Imp_D	0V
Amp_A	0V
Amp_B	0V

Tabla 4-8: Configuración de fuentes de Impedancia y Amplitud.

Debido a esto, la única variación que observaremos durante la simulación del transmisor se deberá solo al patrón de cambios que pondremos en las entradas Eq\_A y Eq\_B que corresponden al control de Pre Énfasis de la línea de transmisión (Figura 3-18).

La configuración para las señales Eq\_A y Eq\_B se muestra en la Tabla 4-9:

Signal	Voltage 1	Voltage 2	Period	Rise Time	Fall Time	Pulse Width
Eq_A	1.8	0	5.32ns	10ps	10ps	2.66ns
Eq_B	1.8	0	10.66ns	10ps	10ps	5.32ns

Tabla 4-9: Configuración de fuentes de entrada Eq\_A y Eq\_B para la evaluación

La frecuencia de las señales van disminuyendo en factor de 2, con el fin de obtener todas las combinaciones posibles entre las dos señales y poder generar la tabla de verdad de datos de prueba, la Tabla 4-10 y la Figura 4-7 muestran las combinaciones de prueba de manera numérica y a nivel señal:

Eq_B	Eq_A
0	0
0	1
1	0
1	1

Tabla 4-10: Combinación numérica de fuentes Eq\_a y Eq\_B

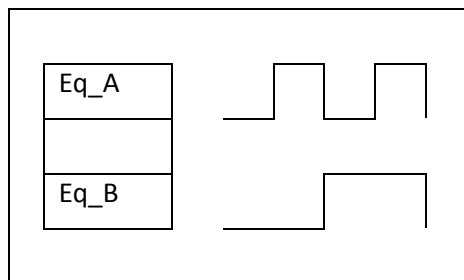


Figura 4-7: Combinación a nivel señal de fuentes Eq\_a y Eq\_B

Haciendo referencia a la Figura 3-18, la configuración de los pines de entrada Data, Random y Data Selector corresponden a cual señal se quiere utilizar como entrada de datos para el transmisor, hágase referencia a la tabla Tabla 4-4 para la configuración de la señal Data, ya que es la misma configuración que se usa en esta sección.

La fuente Random de la Figura 3-18 tiene un elemento al cual se la ha programado una señal de datos aleatoria, la explicación de cómo programar la fuente de datos aleatoria se dará en la sección de la prueba respectiva, la señal Data selector se encarga de seleccionar cuál de estas dos señales entran al Transmisor. Con un valor de 0V en la fuente V1 de la Figura 3-18, la selección de entrada es para Data,

mientras que con un valor de 1.8V en la fuente V1 de la Figura 3-18, la selección de entrada es para la señal Random.

#### 4.3.1 Pruebas del control de pre énfasis:

La Figura 4-8 muestra el resultado de la prueba de control de pre-énfasis de la señal que se desea transmitir, se observa como la señal retardada e invertida al sumarse con la señal original, produce una señal ecualizada para su transmisión:

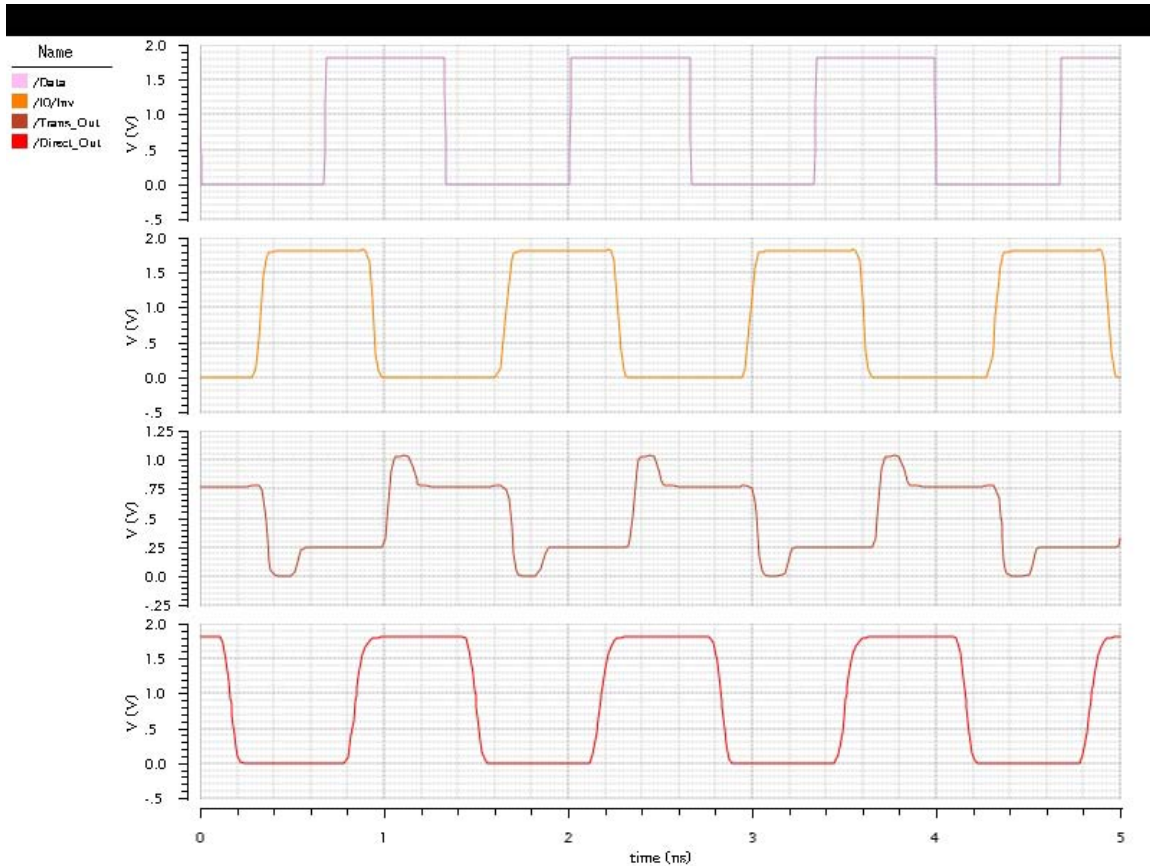


Figura 4-8: Verificación de pre-énfasis

#### 4.4 Setup para probar la señal Random a través del Transmisor:

Nuevamente haciendo referencia a la figura Figura 3-18 para explicar la configuración de las fuentes y señales, el setup de la verificación de señal Random deberá de inhabilitar las variaciones de impedancia, amplitud y ecualización como se muestra en la Tabla 4-11:

Signal	Vdc
Imp_A	0V
Imp_B	0V
Imp_C	0V
Imp_D	0V
Amp_A	0V
Amp_B	0V
Eq_A	0V
Eq_B	0V

Tabla 4-11: Configuración de fuentes de Impedancia, Amplitud y Ecuación.

Debido a esto, la señal Random que se va a utilizar no tendrá ninguna variación del sistema salvo las que pueda ocasionar el transmisor sobre la misma (Figura 3-18).

Haciendo referencia a la Figura 3-18, la configuración de los pines de entrada Data, Random y Data Selector corresponden a la señal se quiere utilizar como entrada de datos para el transmisor, hágase referencia a la tabla Tabla 4-4 para la configuración de la señal Data, ya que es la misma configuración que se usa en esa sección.

La señal Data selector se encarga de seleccionar cuál de estas dos señales entran al Transmisor. Con un valor de 0V en la fuente V1 de la Figura 3-18, la selección de entrada es para Data, mientras que con un valor de 1.8V en la fuente V1 de la Figura 3-18, la selección de entrada es para la señal Random.

#### 4.4.1 Programa Generador de señal Random:

```
# Random sequence of bits
# Generate a random sequence of bits: b_i E {0, 1}. with a length defined
# by the user.
# n_bits = Number of bits
# IU = width of the bit (seconds)
# rf_time = rise time and fall time (seconds)
# VDD = High Voltage (Volts)
# VSS = Low Voltage (Volts)
```

```
import random
```

```
def check_input(in_str):
    try:
        in_num = float(in_str)
```

```

    tst_input = 'ian'
except:
    tst_input = 'nan'
if tst_input == 'nan':
    mult = in_str[len(in_str)-1]
    in_num = float(in_str[0:len(in_str)-1])*mult_dict[mult]
#else:
    # in_num = float(in_str)
return in_num

def bit_voltage(bit):
    if bit > 0:
        bit = VDD
    else:
        bit = VSS
    return(bit)

mult_dict={'T':10**12, 'G':10**9, 'M':10**6, 'K':10**3, 'm':10**(-3), 'u':10**(-6), 'n':10**(-9),
'p':10**(-12), 'f':10**(-15), 'a':10**(-18)}
bits_seq = list()
time_vec = list()
vltg_vec = list()

act_bit = 0
act_time = 0
time_vec.append(act_time)
vltg_vec.append(act_bit)

print 'Random sequence of bits'

n_bits_str = raw_input('number of bits? ')
n_bits = int(n_bits_str)

IU_str = raw_input('IU value [sec]? ')
IU = check_input(IU_str)

rf_time_str = raw_input('Rise/Fall time [sec]? ')

```

```

rf_time = check_input(rf_time_str)

fname = raw_input("Enter file name: ")

VDD_str = raw_input("VDD value [volts]? ")
VDD = check_input(VDD_str)

VSS_str = raw_input("VSS value [volts]? ")
VSS = check_input(VSS_str)

#print n_bits
#print IU
#print rf_time
#raw_input('Enter to continue... ')

for bit in range(1, n_bits+1):
    rndm_bit = random.randint(0,1)
    bits_seq.append(rndm_bit)
    if rndm_bit != act_bit:
        time_vec.append(act_time + rf_time)
        vltg_vec.append(bit_voltage(rndm_bit))
        time_vec.append(act_time + IU)
        vltg_vec.append(bit_voltage(rndm_bit))
        act_bit = rndm_bit
        act_time = act_time+IU
    else:
        time_vec.append(act_time + IU)
        vltg_vec.append(bit_voltage(rndm_bit))
        act_bit = rndm_bit
        act_time = act_time+IU

#print bits_seq
#print time_vec
#print vltg_vec
fhand = open(fname, 'w')
fhand.write("/// This file define the PWL voltage source for a random sequence of bits\n")
fhand.write("/// This is random sequence of bits:\n")

```

```

fhand.write('/// \n')
fhand.write('/// ')
fhand.write(str(bits_seq))
fhand.write('\n')
fhand.write('///\n')
fhand.write('/// IU Value = '+str(IU))
fhand.write('\n')
fhand.write('/// Rise/Fall times = '+str(rf_time))
fhand.write('\n')
fhand.write('/// VDD value = '+str(VDD))
fhand.write('\n')
fhand.write('/// VSS value = '+str(VSS))
fhand.write('\n')
fhand.write('/// Here is the PWL source definition...\n')
for point in range(0, len(time_vec)):
    fhand.write(str(time_vec[point]))
    fhand.write(' \t')
    fhand.write(str(vltg_vec[point]))
    fhand.write('\n')
fhand.write('/// End of definition')
fhand.close()

```

Este programa fue desarrollado en **Python** [9] con el propósito de generar una señal aleatoria para probar dispositivos que manejan información digital. El programa es configurable para que nos arroje un archivo de datos aleatorios al cual le definimos el número de datos, el periodo de los mismos, el raise y fall time, el nivel de voltaje de las señales de referencia Vdd y Vss y el nombre que se le quiere dar al archivo.

Este archivo en conjunto con una fuente VPWLF, nos genera la señal Random de prueba para el sistema, solo hay que ponerle la ruta en la que se encuentra el archivo en el campo PWL file name de la fuente.

#### 4.4.2 Pruebas del circuito Inversor

Las Figura 4-9 y la Figura 4-10 presentan el test-bench y los resultados de simulación del circuito inversor.



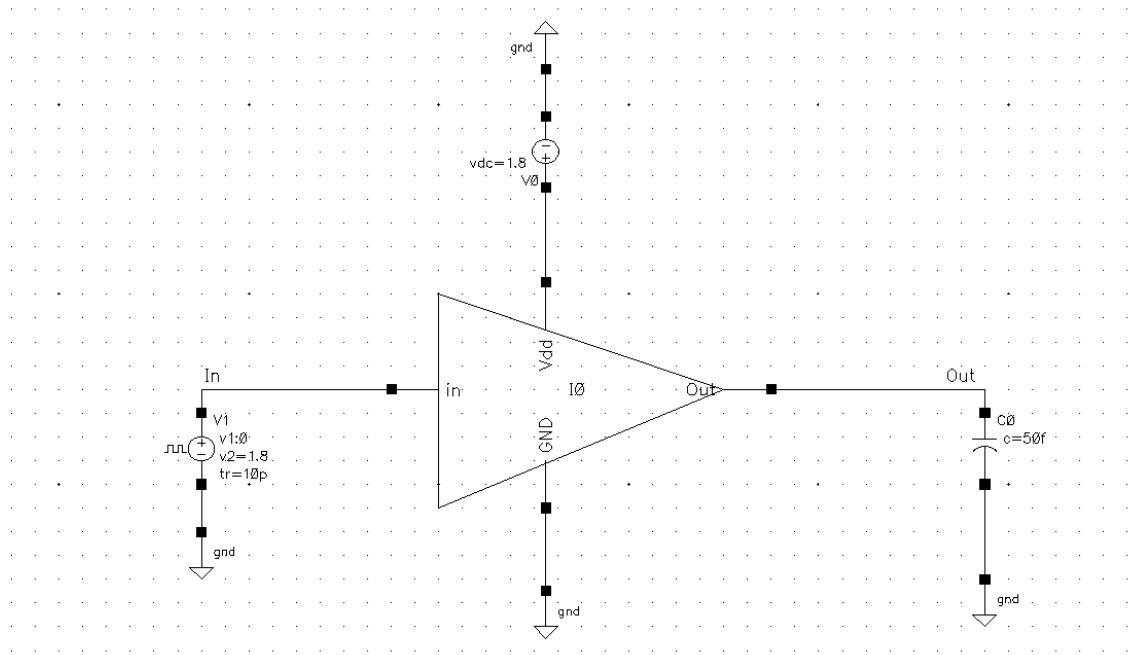


Figura 4-9: Diagrama Test Bench para el inversor

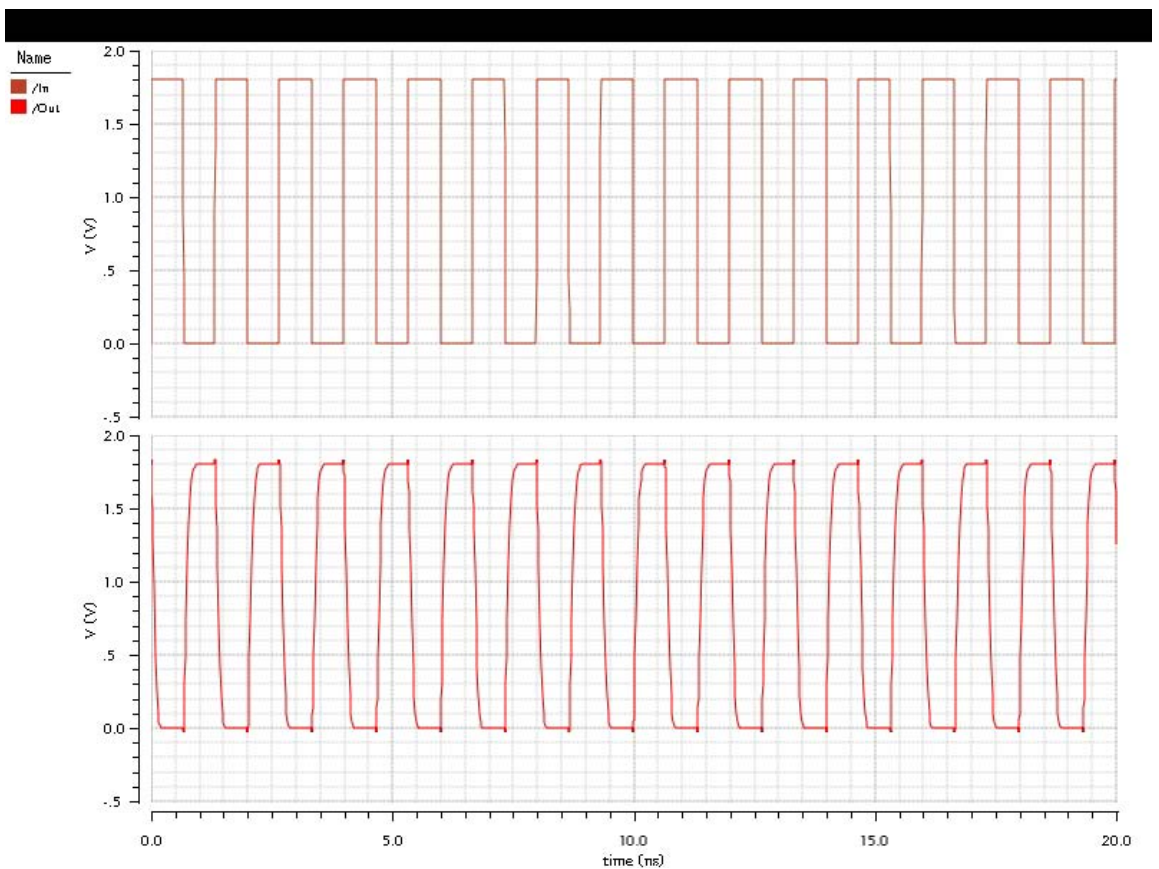


Figura 4-10: Respuesta del Inversor

En la Figura 4-10 se observa la inversión de la señal al pasar por el circuito inversor.

En general para el diseño de compuertas digitales se toman en cuenta las siguientes consideraciones; Se obtiene una característica de transferencia simétrica

cuando los dispositivos se diseñan para que tengan parámetros de transconductancia iguales, una condición llamada *coincidencia*. Como la  $\mu_n$  es dos a cuatro veces la  $\mu_p$ , la coincidencia se logra haciendo  $(W/L)_p$  dos a cuatro veces la  $(W/L)_n$ .

Normalmente los dos dispositivos tienen la misma longitud de canal  $L$ , la cual se establece al mínimo admisible para la tecnología del proceso dada. El ancho mínimo del transistor NMOS es de una y media a dos veces  $L$ , y el ancho del transistor PMOS de dos a tres veces  $L$ .

#### 4.4.3 Pruebas de compuerta NOR de dos entradas (NOR2)

Las Figura 4-11 y Figura 4-12 presentan el test-bench y los resultados de simulación del circuito NOR de dos entradas.

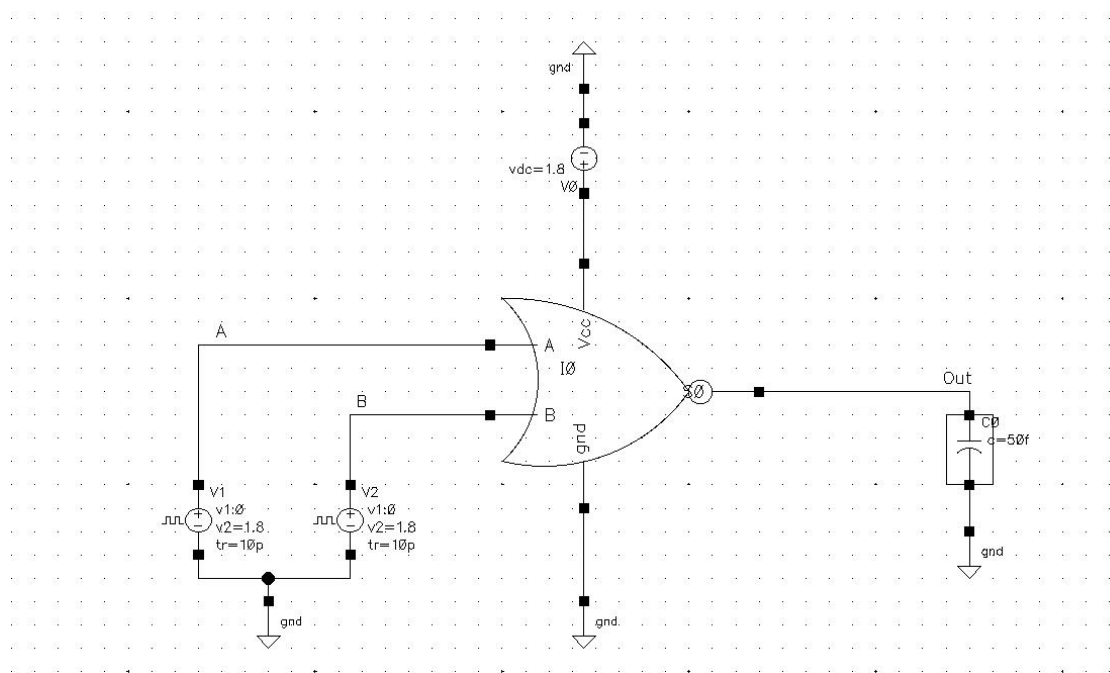


Figura 4-11: Test bench del circuito de compuerta NOR2

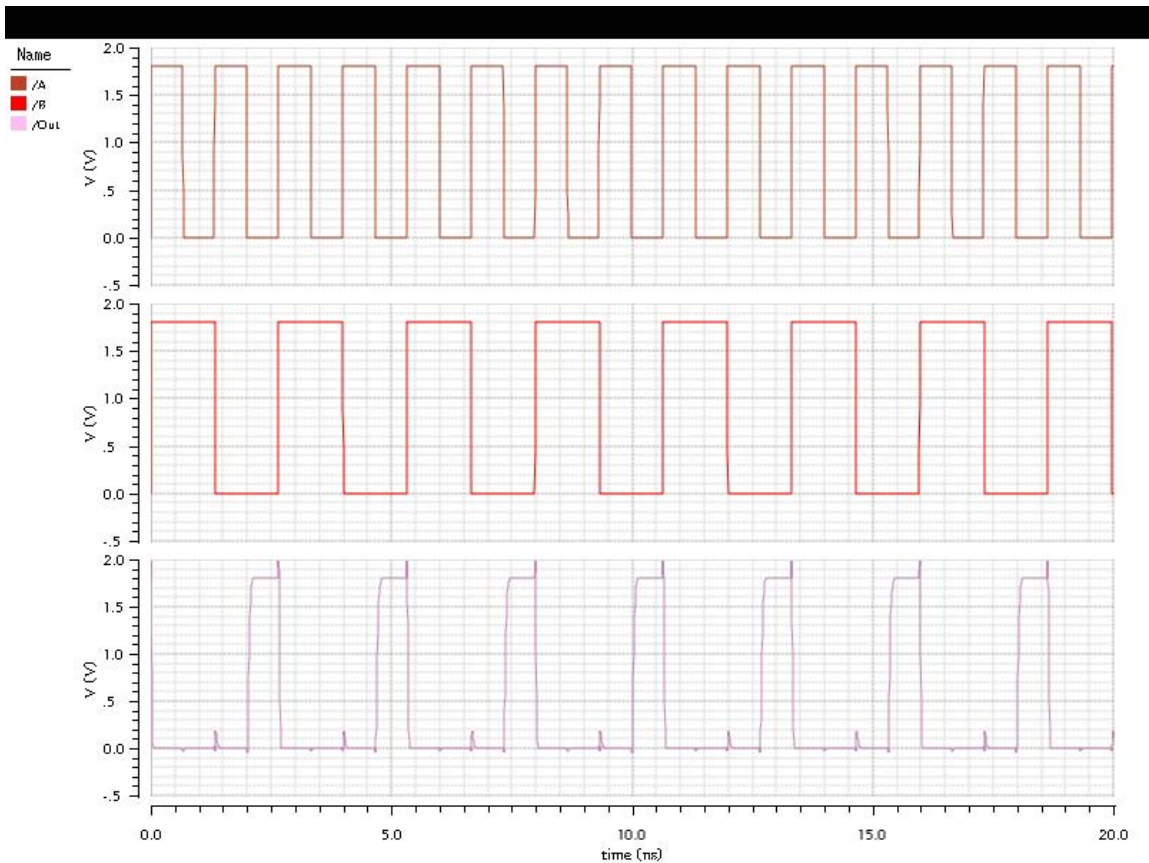


Figura 4-12: Respuesta transitoria del circuito de Compuerta NOR2

Se observa la tabla de verdad de la compuerta a través de las combinaciones de las señales digitales de entrada y su respuesta.

#### 4.4.4 Pruebas de NAND de dos entradas (NAND2)

Las Figura 4-13 Figura 4-14 presentan el test-bench y los resultados de simulación del circuito NAND de dos entradas.

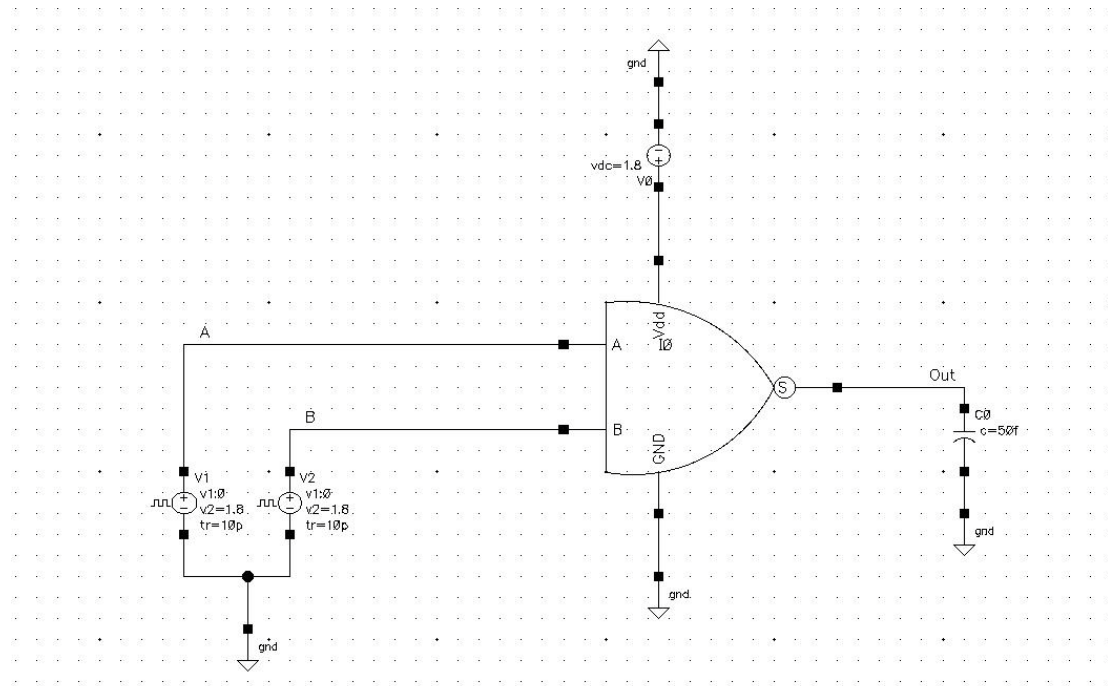


Figura 4-13: Test Bench para simular el circuito de compuerta NAND2

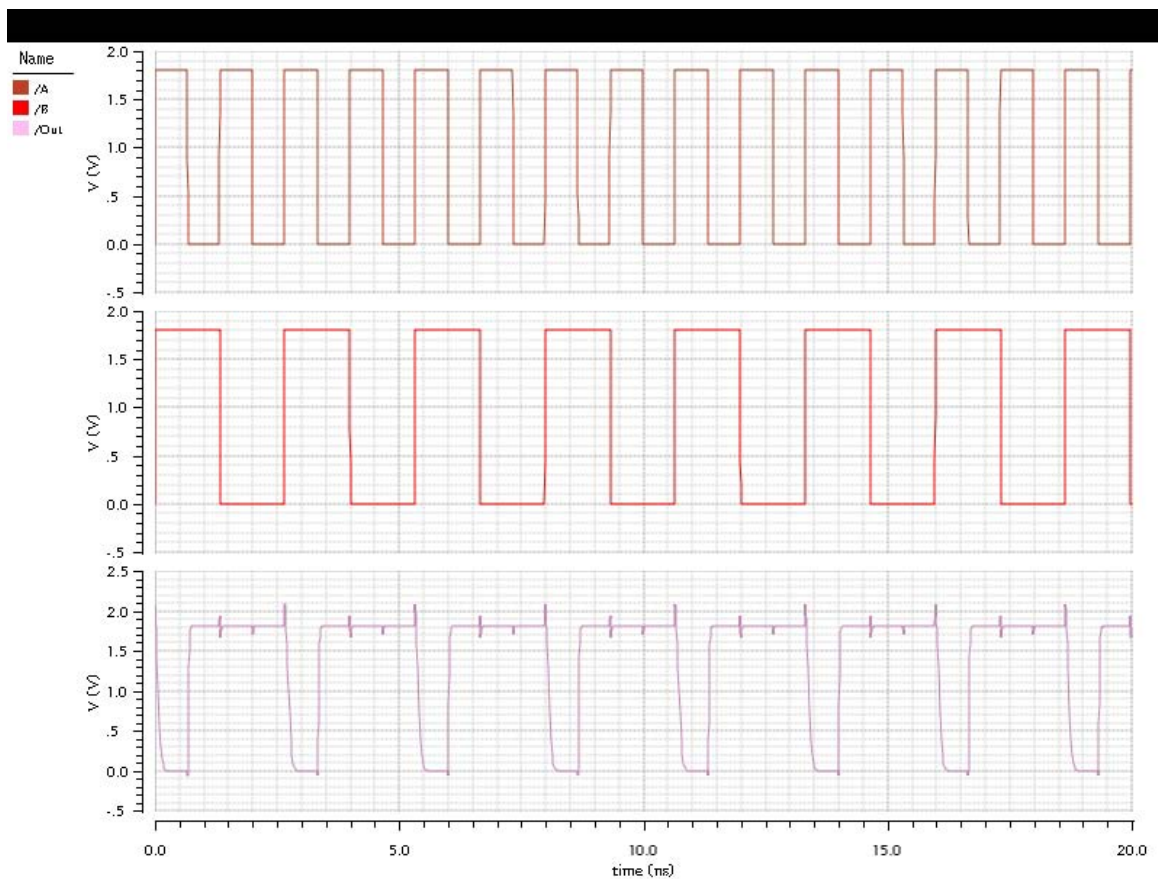


Figura 4-14: Respuesta transitoria del circuito de compuerta NAND2

Podemos observar la tabla de verdad de la compuerta a través de las combinaciones de las señales digitales de entrada y su respuesta.



#### 4.4.5 Pruebas compuerta NOR de tres entradas (NOR3)

Las Figura 4-15 y Figura 4-16 presentan el test-bench y los resultados de simulación del circuito NOR de tres entradas:

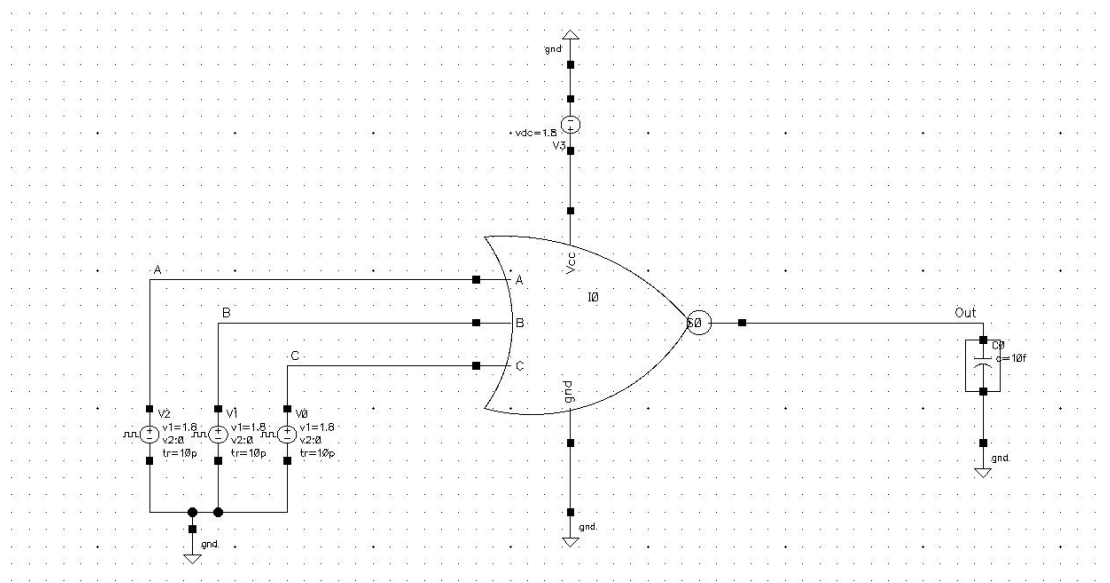


Figura 4-15: Test bench para simular el circuito de compuerta NOR3

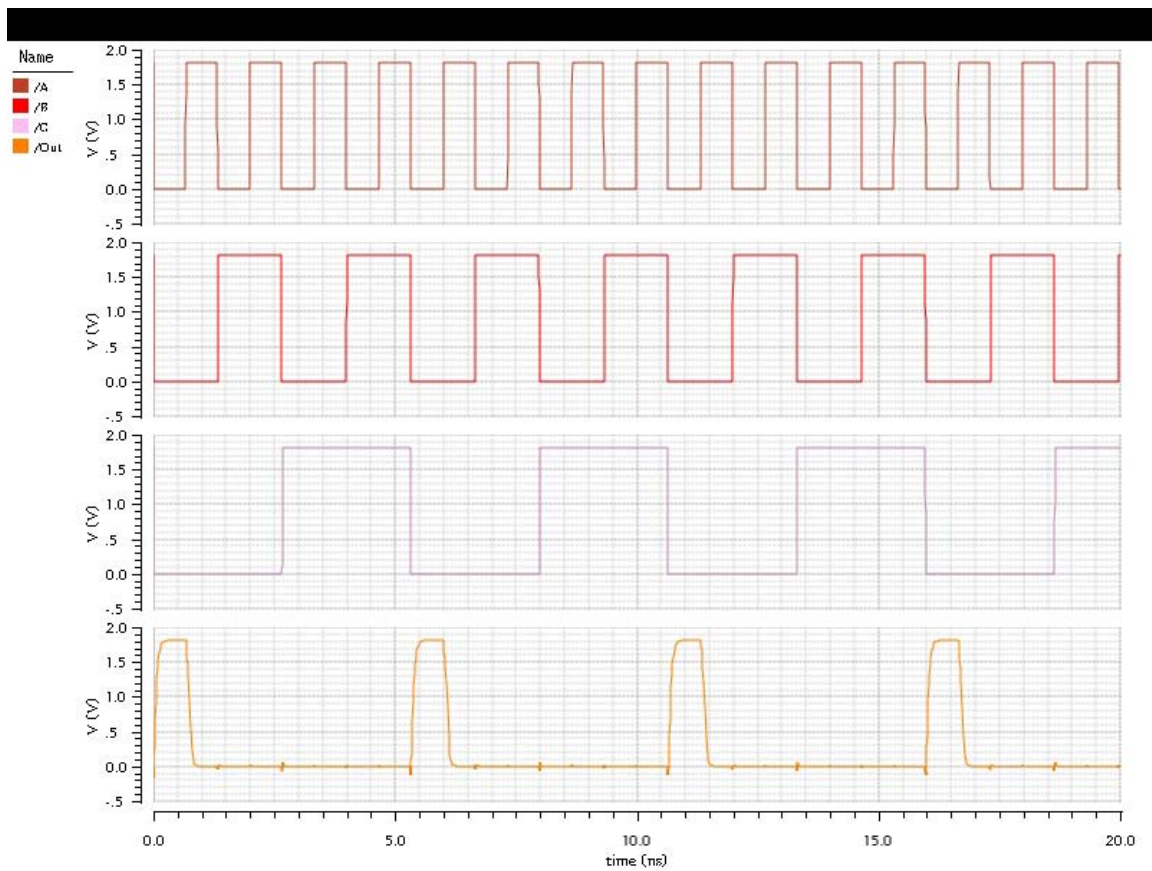


Figura 4-16: Respuesta transitoria del circuito de compuerta NOR3

Podemos ver la tabla de verdad de la compuerta con las combinaciones de entradas y la respuesta del sistema.

#### 4.4.6 Pruebas compuerta NAND de tres entradas (NAND3)

Las Figura 4-17 y Figura 4-18 presentan el test-bench y los resultados de simulación del circuito NAND de tres entradas:

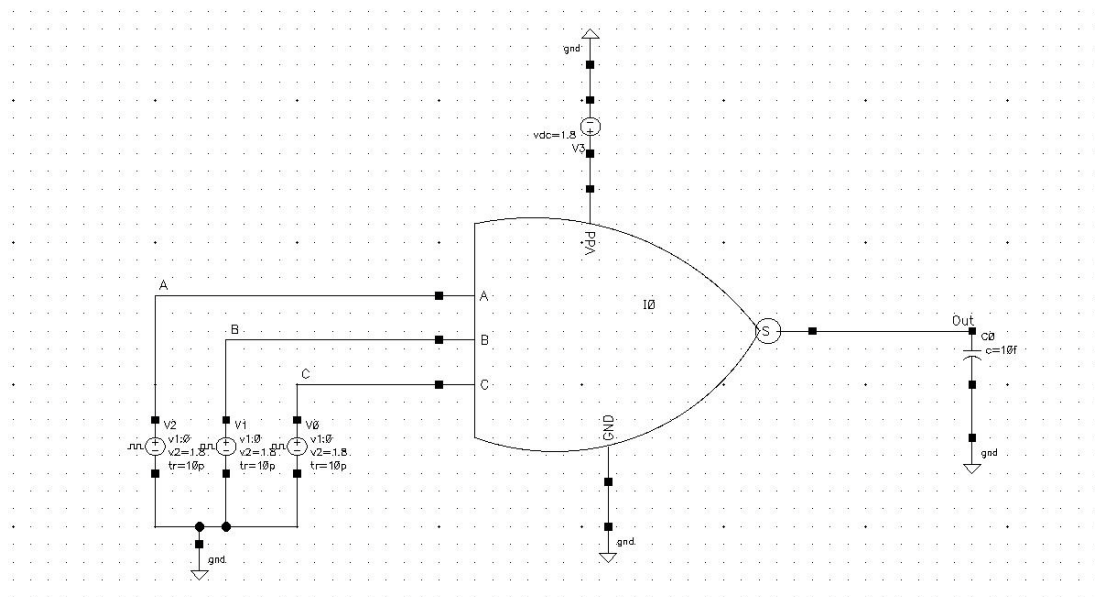


Figura 4-17: Test bench para simular el circuito de compuerta NAND3

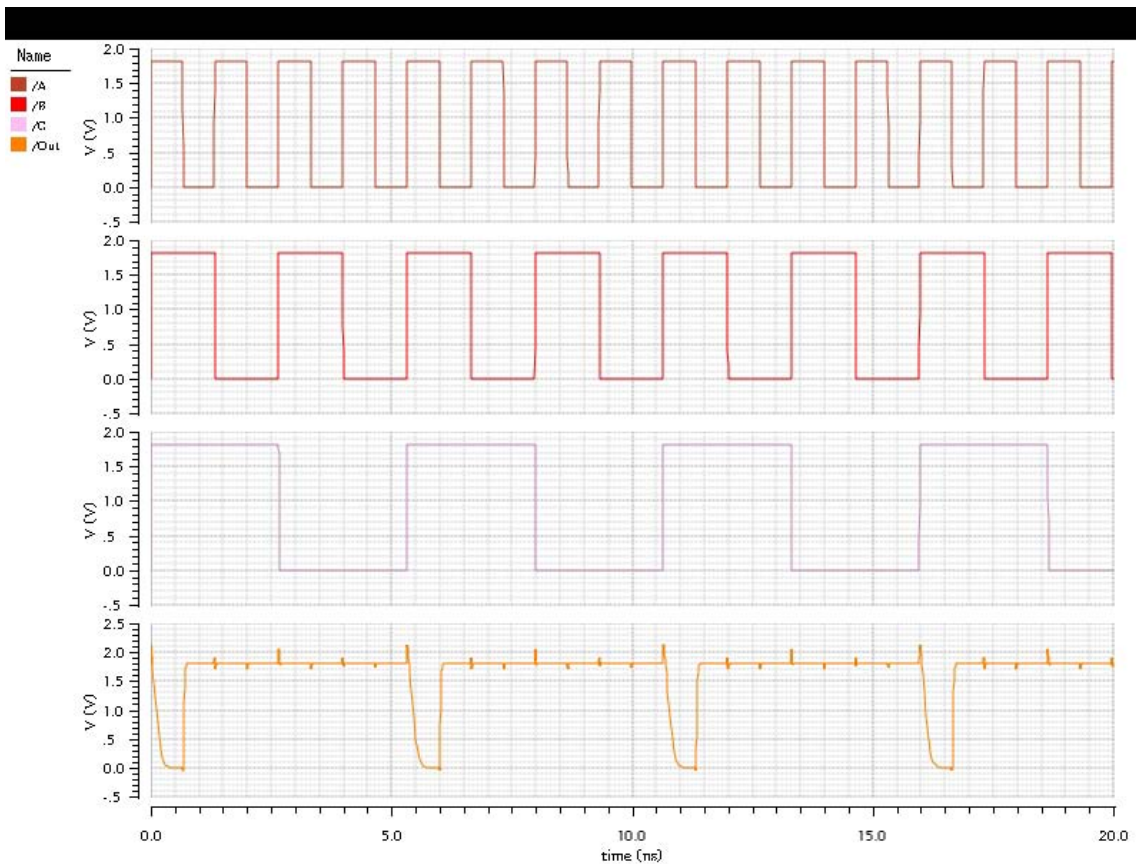


Figura 4-18: Respuesta de compuerta NAND de tres entradas

Podemos ver la tabla de verdad de la compuerta con las combinaciones de entradas y la respuesta del sistema.

#### 4.4.7 Pruebas del Buffer de tres estados

Las Figura 4-19 y Figura 4-20 presentan respectivamente el esquemático testbench y resultados de simulación de la compuerta del buffer de tres estados:

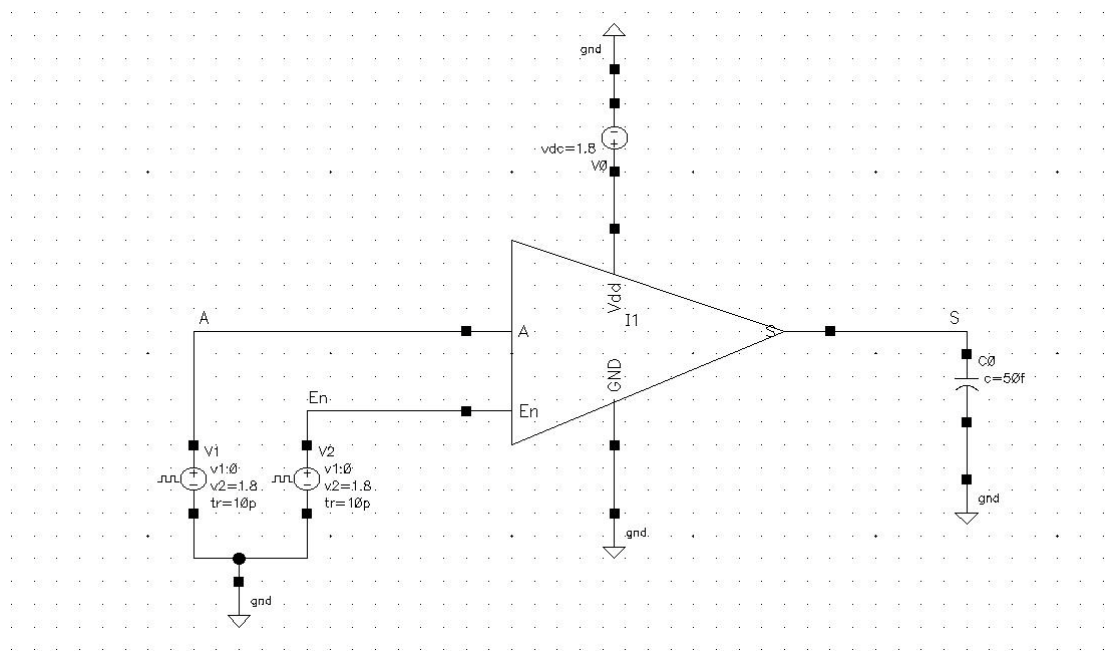


Figura 4-19: Test bench del circuito Buffer de tres estados

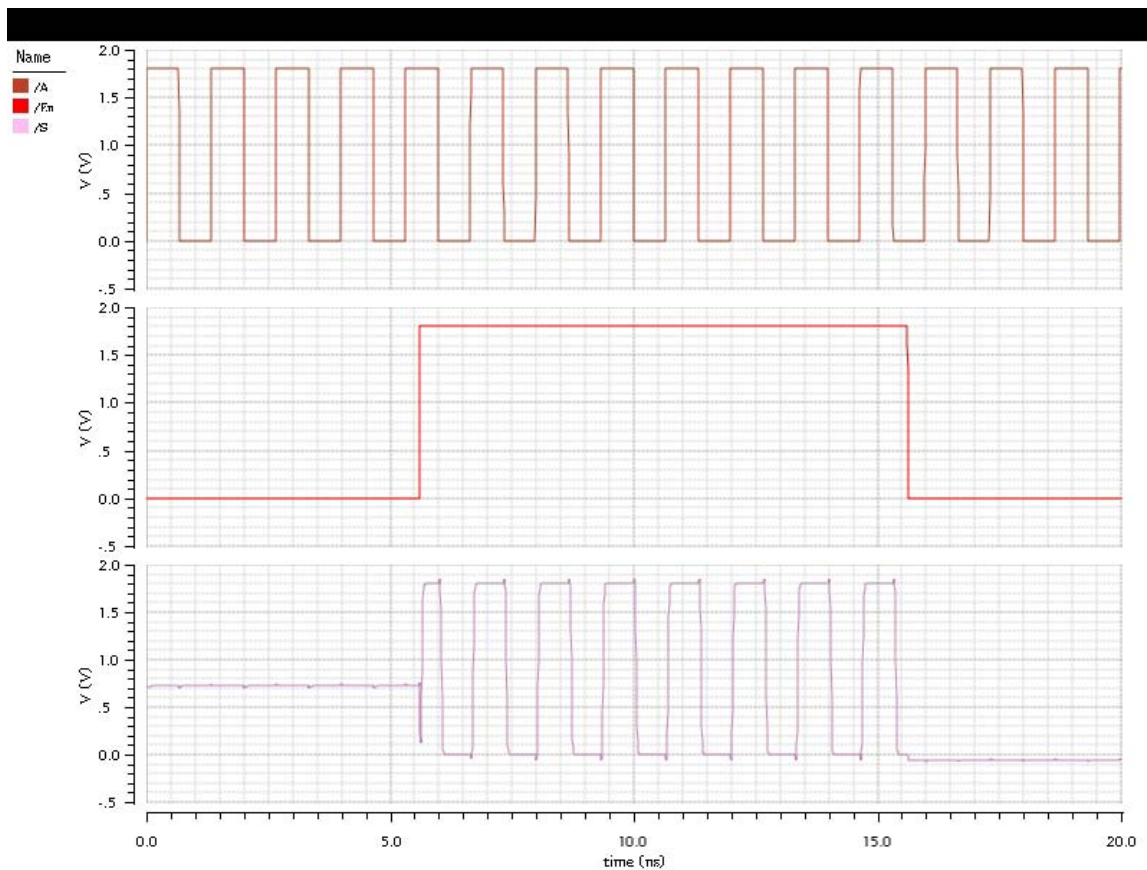


Figura 4-20: Respuesta transitoria del circuito Buffer de tres estados

Podemos observar como el Buffer permanece en tercer estado cuando no se encuentra habilitado, y permite pasar la señal cuando se activa su *enable*, dejando pasar tanto unos como ceros. Una vez desactivado, regresa al estado de alta impedancia.

#### 4.4.8 Pruebas del Inversor-Retardador

Las Figura 4-21 y Figura 4-22 presentan el test bench y los resultados de simulación del Inversor-Retardador

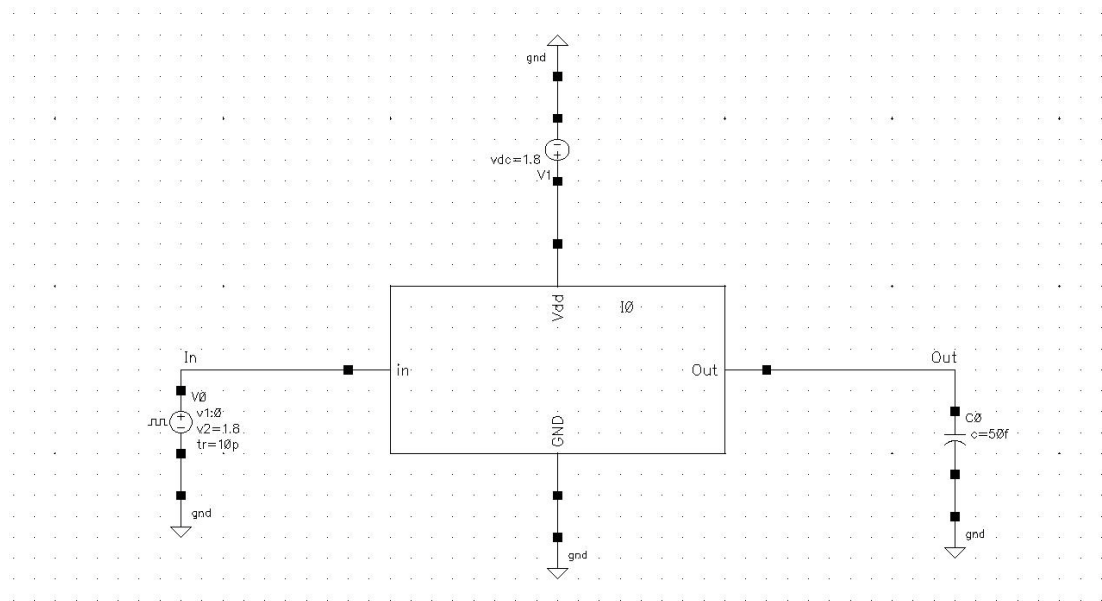


Figura 4-21: Test bench del circuito Retardador-Inversor





Figura 4-22: Respuesta del Buffer Retardador-Inversor

Se puede observar como el Buffer invierte la señal y la retrasa un 10 % aproximadamente como se esperaba.

#### 4.4.9 Pruebas del decodificador de 4 entradas y 12 salidas, (DECO 4 a 12)

Integramos los 3 sub-módulos del decodificador para validar la respuesta a nivel de simulación pre-layout de este circuito. La Figura 4-23 presenta el test bench del DECO 4 a 12.

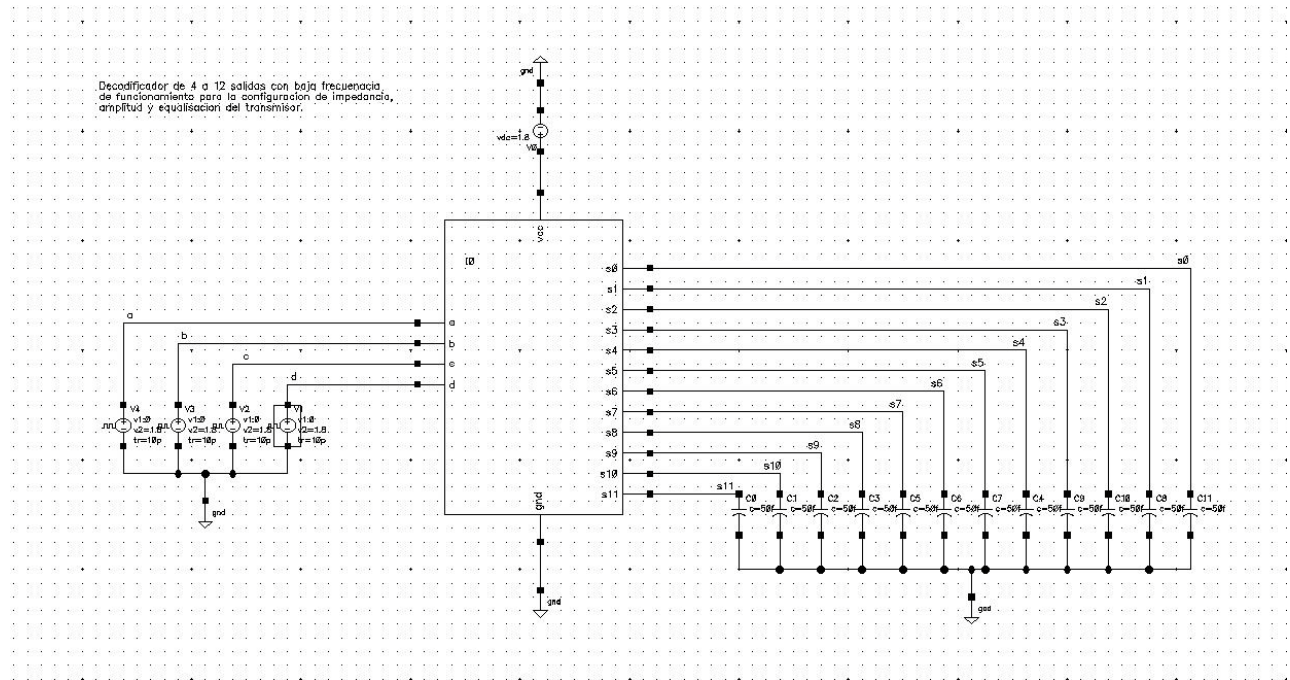


Figura 4-23: Test Bench para decodificador de 4 a 12.

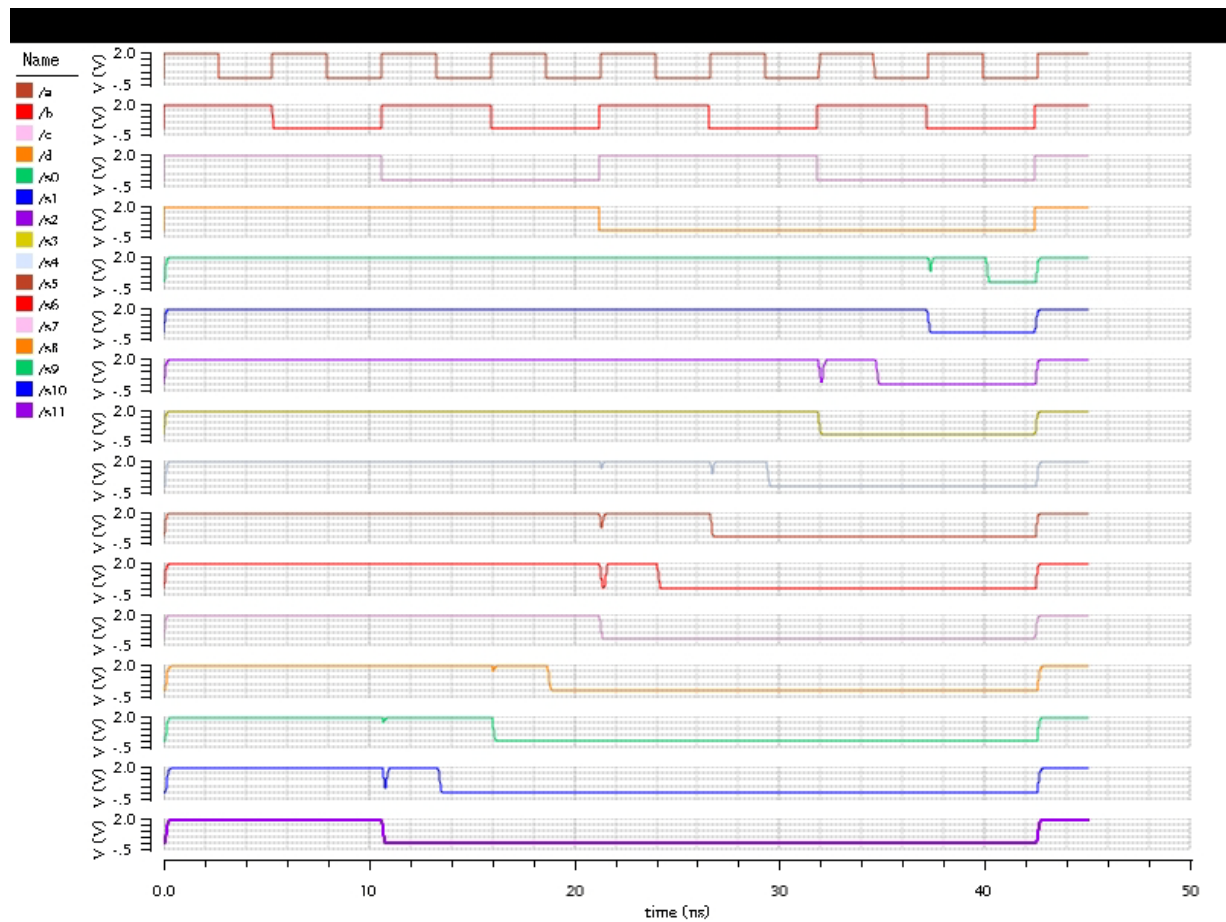


Figura 4-24: Respuesta del Decodificador 4 entradas 12 salidas

La Figura 4-24 presenta los resultados del decodificador de 4 entradas 12 salidas, se observa como para las diferentes combinaciones de bits de entrada, el decodificador

tiene un comportamiento de salida tipo termómetro, en donde las señales se van poniendo en uno lógico en secuencia de lsb al msb.

#### 4.4.10 Pruebas del decodificador de 2 entradas y 3 salidas, (DECO 2 a 3)

Las Figura 4-25 y Figura 4-26 presentan respectivamente el test-bench y los resultados de simulación del circuito DECO 2 a 3.

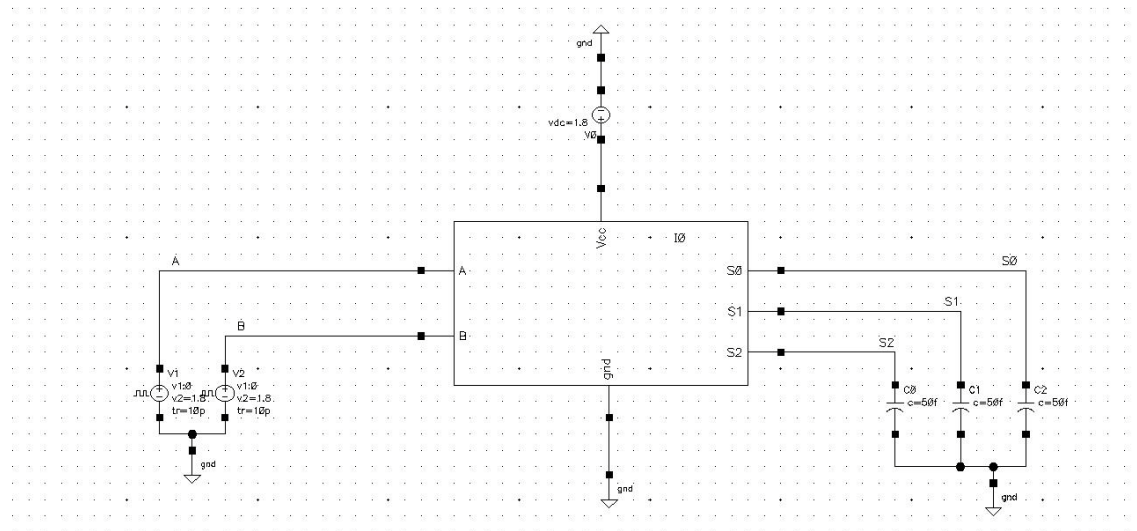


Figura 4-25: Decodificador 2 entradas 3 salidas test bench

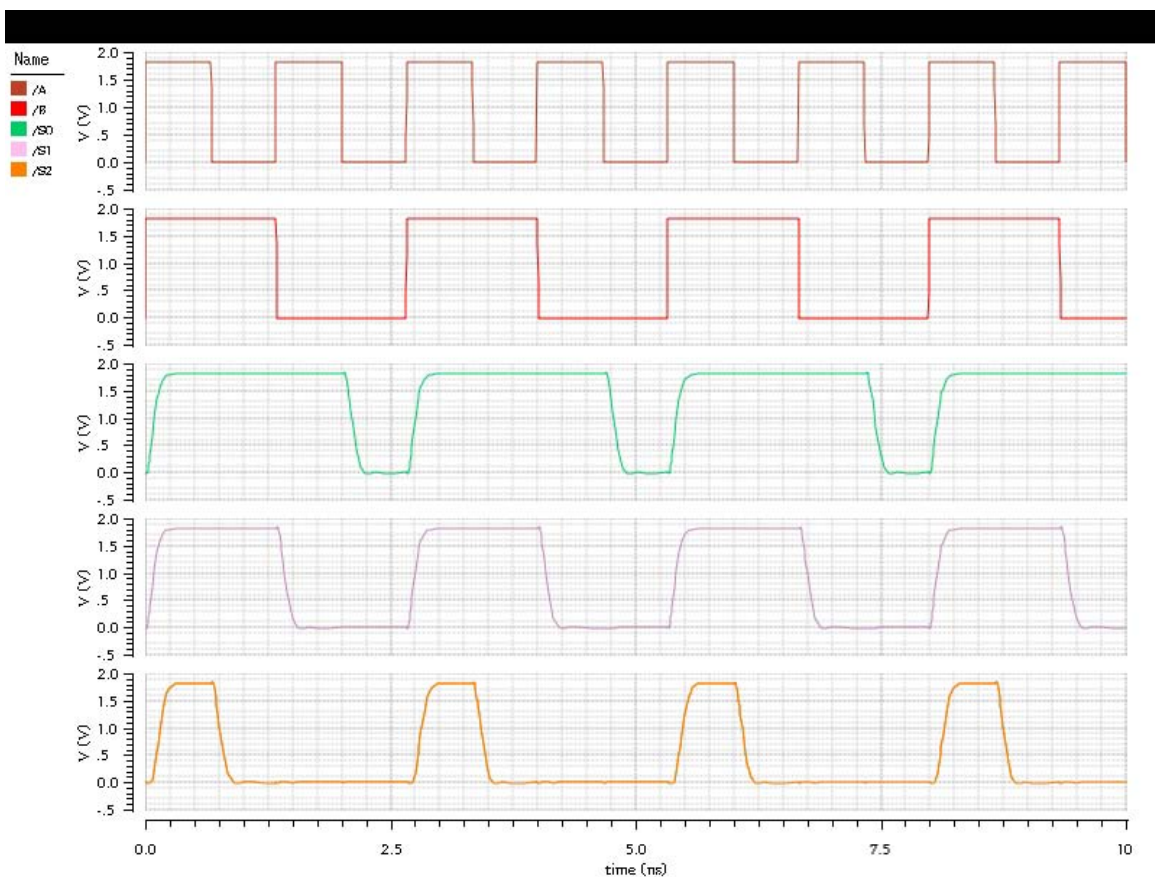


Figura 4-26: Respuesta del decodificador 2 entradas 3 salidas

La Figura 4-26 presenta los resultados del decodificador de 2 entradas 3 salidas, se observa como para las diferentes combinaciones de bits de entrada, el decodificador tiene un comportamiento de salida tipo termómetro, en donde las señales se van poniendo en uno lógico en secuencia de lsb al msb.

#### 4.4.11 Pruebas del Multiplexor (Mux 2 a 1)

Las Figura 4-27 y Figura 4-28 presentan el test-bench y los resultados de simulación del circuito MUX de 2 a 1:

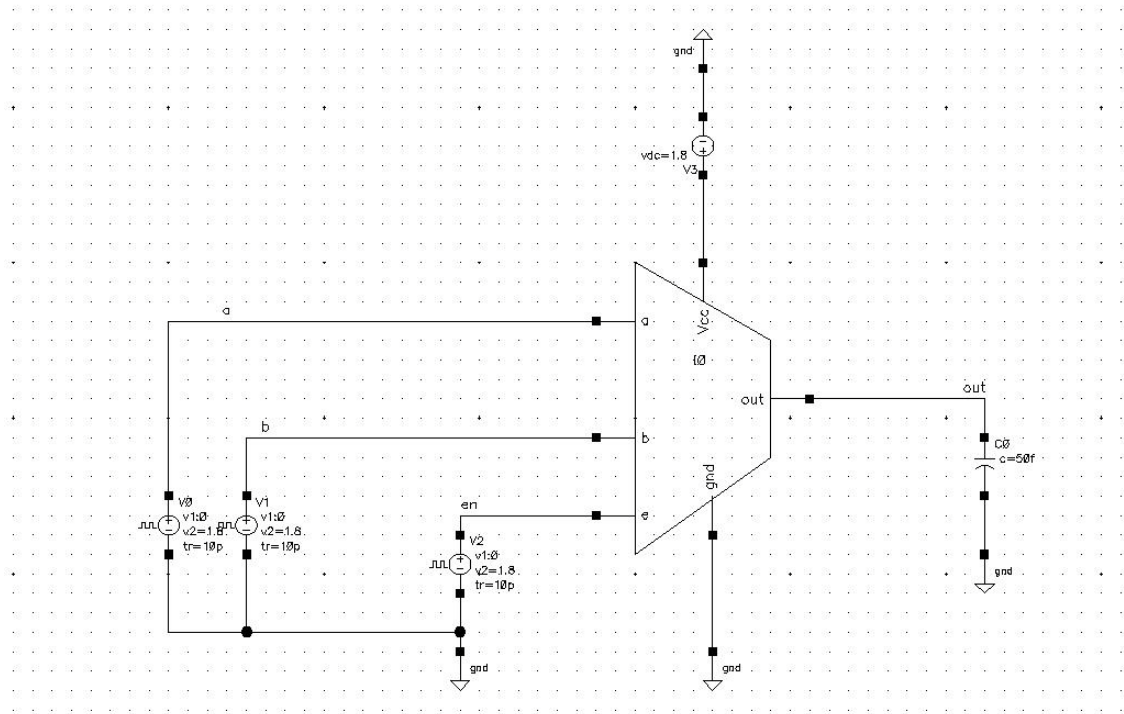


Figura 4-27: Test-bench del circuito Mux 2 a 1.



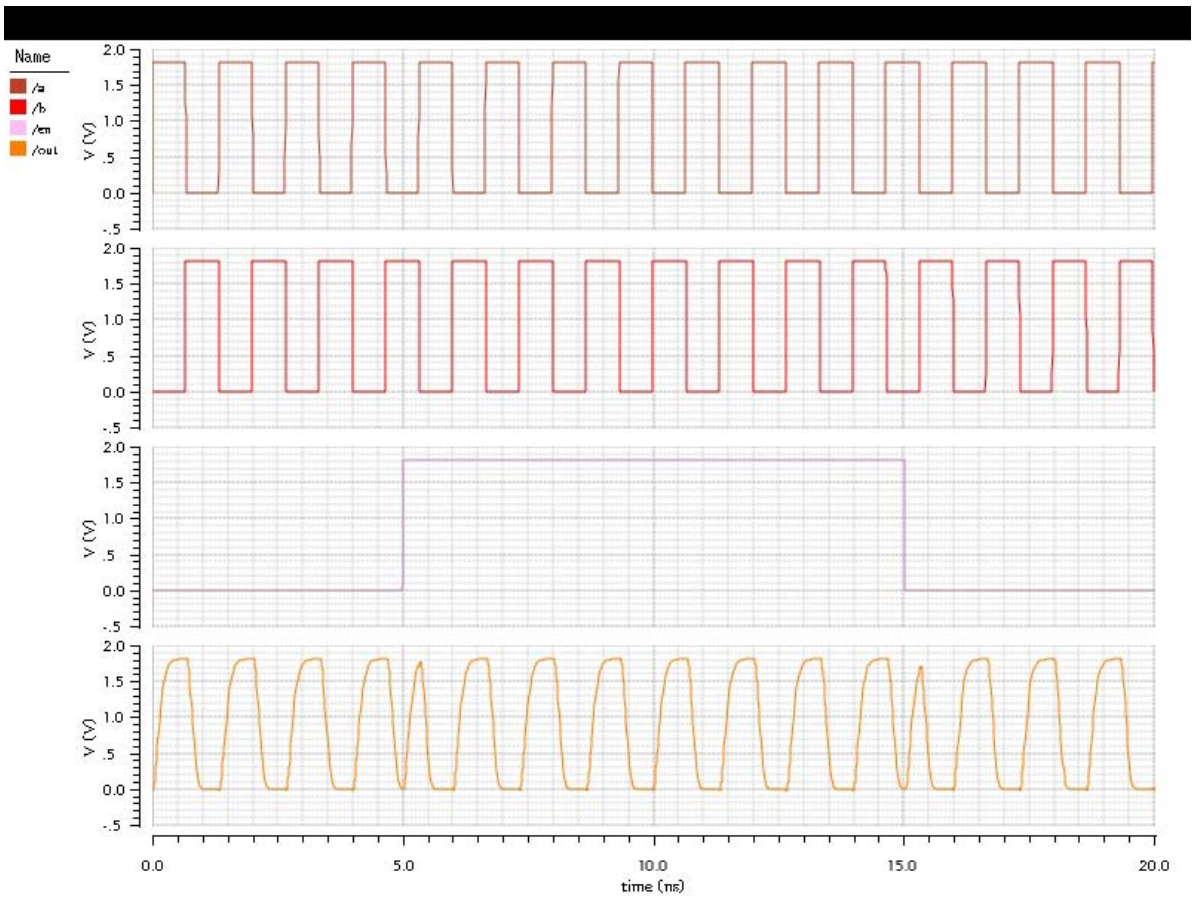


Figura 4-28: Respuesta transitoria del Mux 2 a 1

En la Figura 4-28 se observa como el multiplexor cambia de una señal a otra dependiendo del estado del selector

#### 4.4.12 Pruebas del circuito amarrador (tied)

Las Figura 4-29 y Figura 4-30 presentan el test-bench y los resultados de simulación del circuito amarrador.

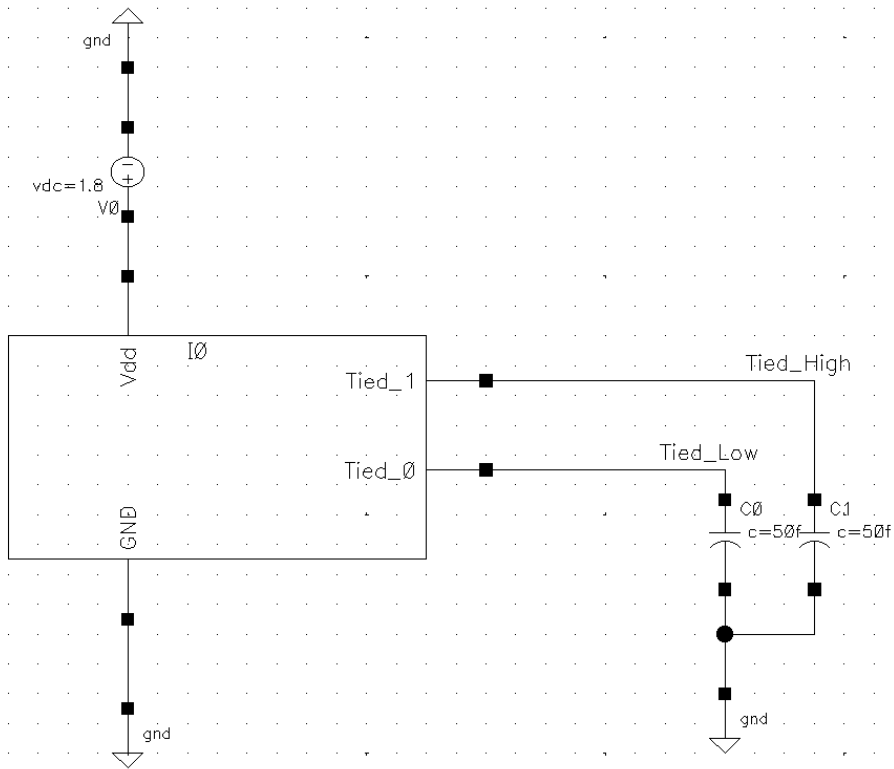


Figura 4-29: Test bench del circuito Amarrador (Tied)

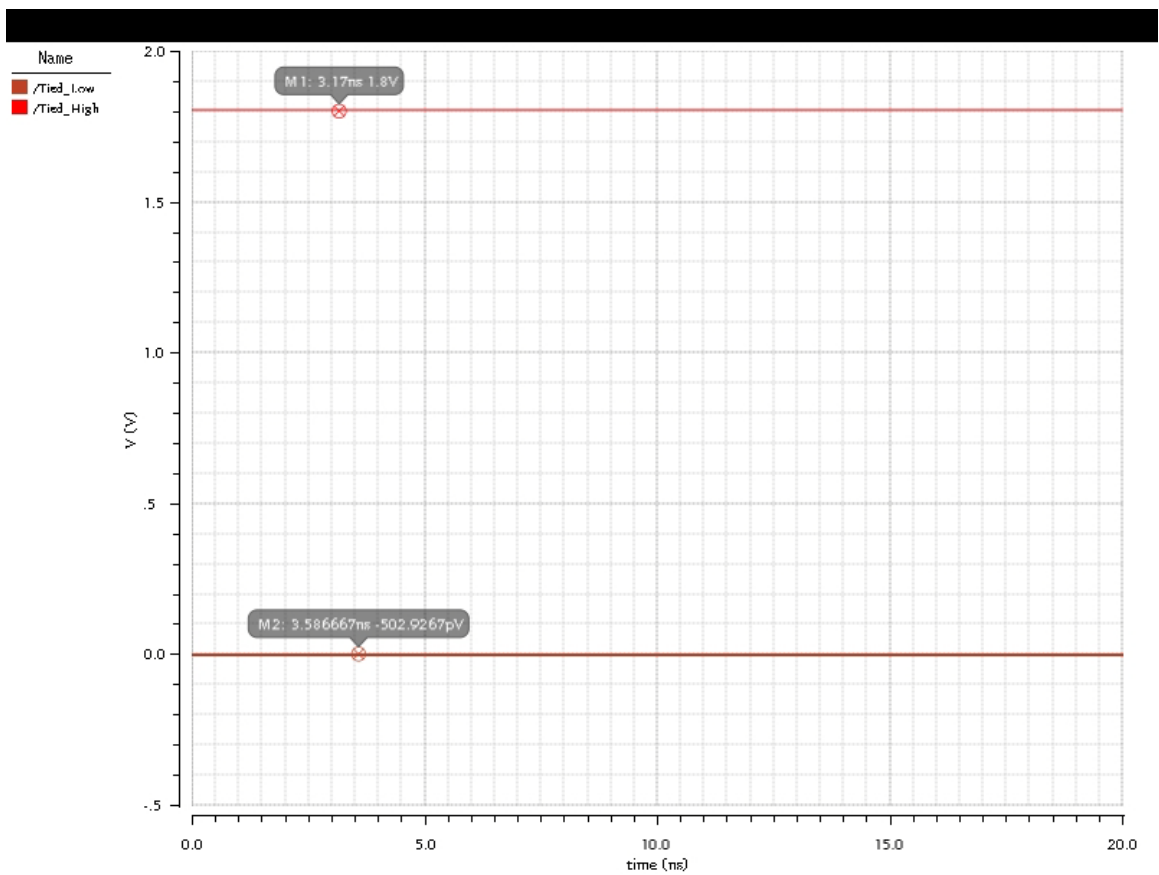


Figura 4-30: Respuesta transitoria del circuito Amarrador (Tied)



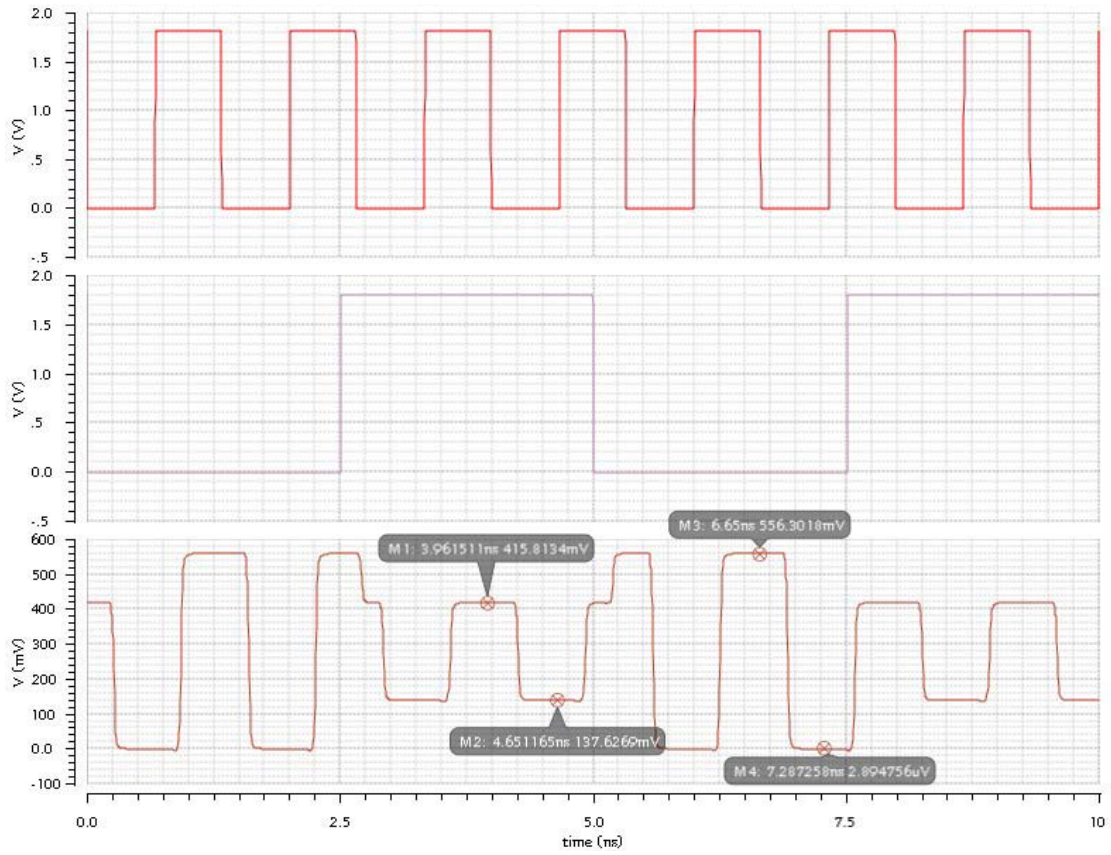


Figura 4-32: Evaluación de cambio de amplitud.

La verificación del control de impedancias se muestra en la Figura 4-33:

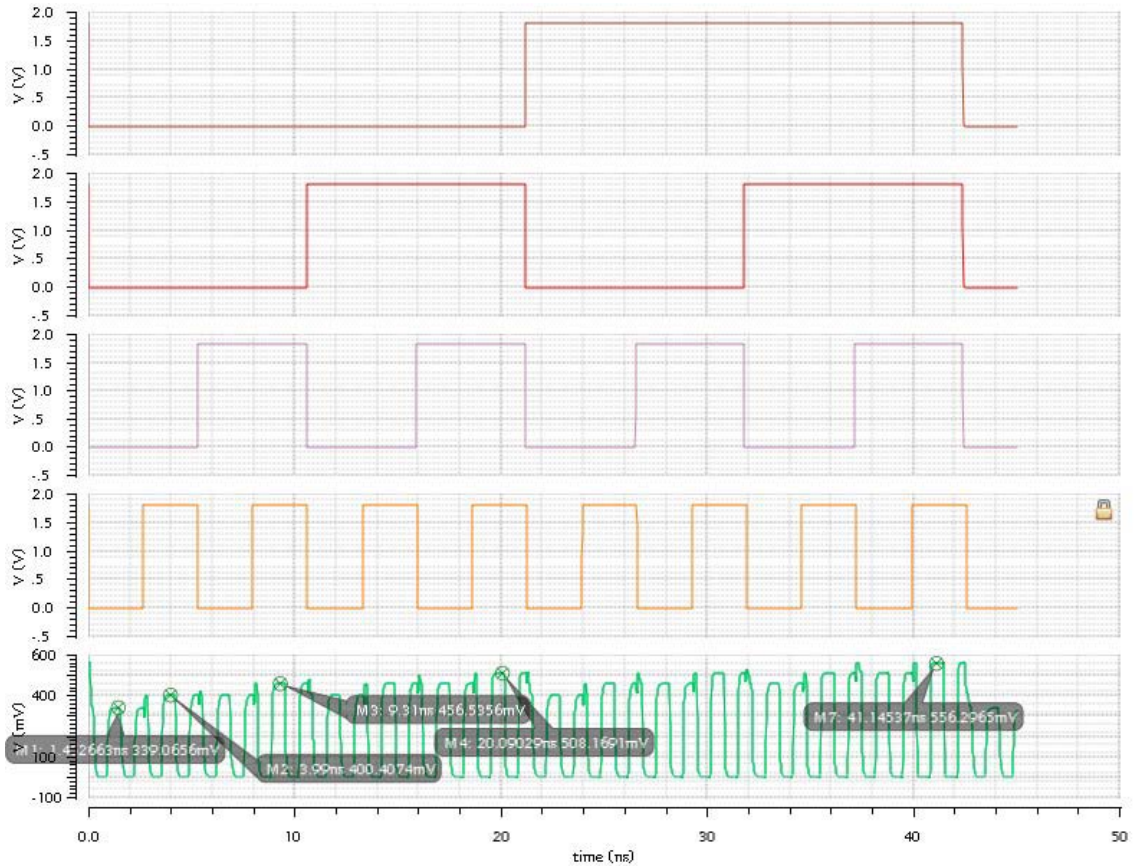


Figura 4-33: Evaluación de cambio de impedancia.



Se observa a través del cambio de amplitud de la señal de salida (curva en color verde en la Figura 4-33) que la impedancia del sistema está cambiando. Considerando que la resistencia de carga es de  $100\ \Omega$ , podemos saber la impedancia del sistema en cada caso, empleando la Ecuación 4-1 que viene del divisor de voltaje con la resistencia  $R_B$  despejada:

Tenemos los siguientes datos:

$$V_{in} = 1.8\ V$$

$$R_A = 100\ \Omega$$

Para un voltaje de salida de  $V_{out} = 339.065\ mV$  nos da una impedancia igual a  $R_B = 430\ \Omega$

Para un voltaje de salida de  $V_{out} = 556.29\ mV$  nos da una impedancia igual a  $R_B = 224\ \Omega$

Los valores de impedancia de salida son más elevados de lo esperado (alrededor de  $100\ \Omega$ ) debido a que falta tomar en cuenta el efecto de los otros dos módulos ZAP conectados en paralelo.

La verificación del pre-énfasis se muestra en la Figura 4-34:

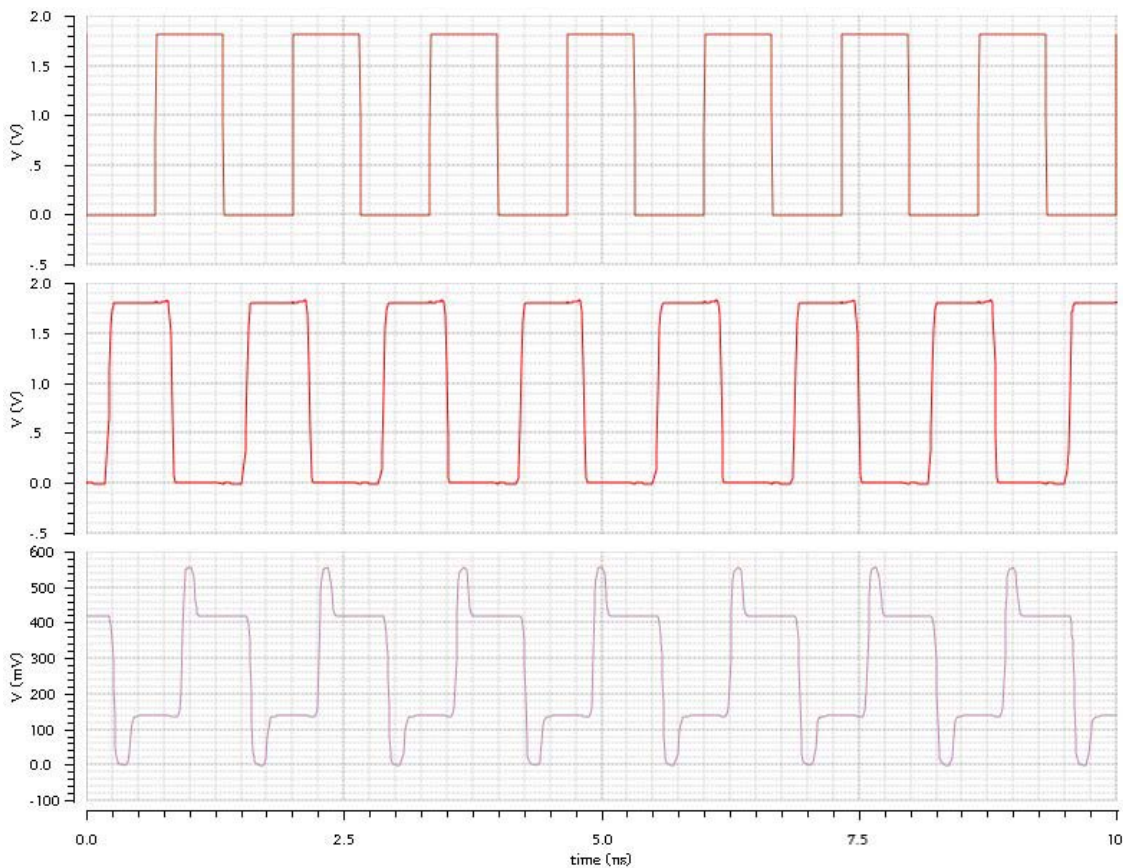


Figura 4-34: Evaluación de pre-énfasis.

Se observa como las dos señales se combinan y forman la señal de pre-énfasis.

El transmisor de Amplitud, Impedancia y pre-Énfasis es un dispositivo muy versátil que nos ayuda a adaptar nuestro SerDes a varias situaciones de interconexión con varios sistemas que requieren intercomunicación serial.

Las pruebas realizadas al Transmisor mostraron buenos resultados en el control de impedancia, pre-énfasis y amplitud de los datos transmitidos.



## 5 Diseño del Layout del módulo Transmisor de datos del sistema SerDes

El circuito se diseñó en tecnología CMOS de 180nm (proceso IBM cmrf7sf) con licencia de MOSIS. En este capítulo se documenta el diseño de layout full custom de las celdas básicas que componen el módulo Transmisor.

El diagrama a bloques del circuito descrito en el párrafo anterior se muestra en la Figura 5-1:

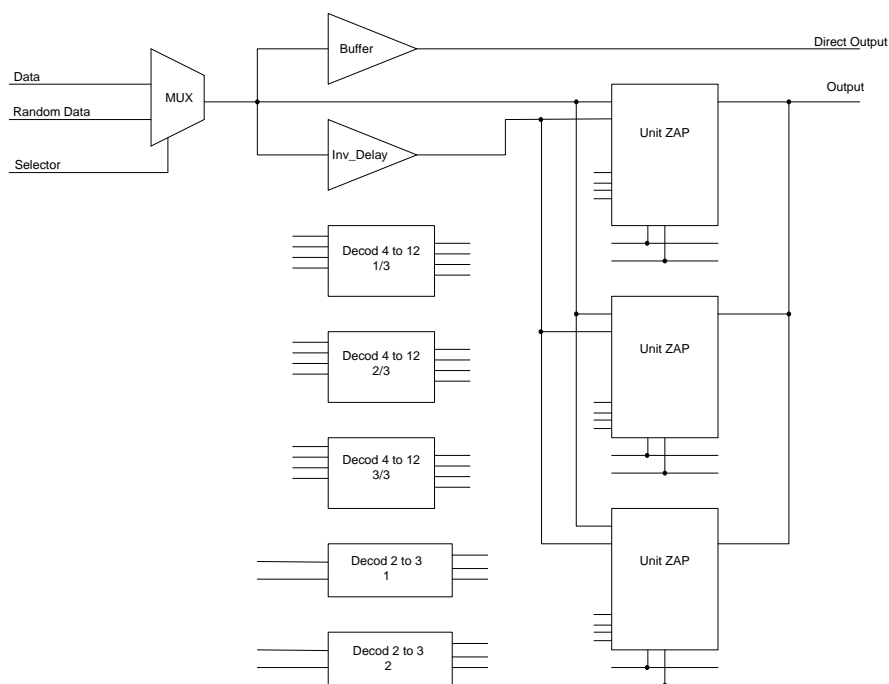


Figura 5-1: Diagrama a bloques del circuito transmisor de datos [5].

## 5.1 Primera Aproximación del Floorplan

Las siguientes tablas presentan una estimación de área que requiere cada uno de los bloques que componen el módulo transmisor.

Bloque	Largo ( $\mu\text{m}$ )	Ancho ( $\mu\text{m}$ )	Area ( $\mu\text{m}^2$ )
NAND 3	8.88	4.64	41.20
NOR 3	8.24	3.4	28.02
Buffer	6.08	6.09	37.03
Inverter	4.35	1.96	8.53
MUX	7.09	15.04	106.63
NAND 2	8.44	2.64	22.28
NOR 2	8.89	2.68	23.83
Delay block	5.91	10.84	64.06
Tied	7.01	5.86	41.08
Resistor 820 $\Omega$	274	8	2.19e3
3 States Buffer	13.77	13.58	187.00

Tabla 5-1: Dimensiones de los bloques funcionales.

Unidad ZAP		
Bloque	Cantidad	Area ( $\mu\text{m}^2$ )
3 States Buffer	8	1495.97
8 Resistor (820 $\Omega$ )	8	17536.00
Buffer	5	185.14
MUX	4	426.53
Tied	1	41.08
Total		19684.72

Tabla 5-2: Dimensión bloque de unidad ZAP.

Decodificador 4 - 12		
Bloque	Cantidad	Area ( $\mu\text{m}^2$ )
Inverter	26	221.68
NOR 3	4	112.06
NAND 3	2	82.41
NOR 2	7	166.78
NAND 2	11	245.10
Total		828.02

Tabla 5-3: Dimensión bloque decodificador 4-12.

<b>Decodificador 2 - 3</b>		
<b>Bloque</b>	<b>Cantidad</b>	<b>Area (µm²)</b>
Inverter	4	34.10
NOR 2	1	23.83
NAND 2	1	22.28
Total		80.21

Tabla 5-4: Dimensión bloque decodificador 2-3.

<b>Circuito Total 1 CH</b>		
<b>Bloque</b>	<b>Cantidad</b>	<b>Area (µm²)</b>
Bloque ZAP	3	59054.17
Decod 4-12	1	828.02
Decod 2-3	2	160.42
Invert-Delay	1	64.06
Buffer	1	37.03
MUX	1	106.63
Total		60250.33

Tabla 5-5: Dimensión bloque del circuito total de 1 canal.

<b>Circuito Total 1 Ch Inv</b>		
<b>Bloque</b>	<b>Cantidad</b>	<b>Area (µm²)</b>
Circuito Total 1 Ch	1	60250.33
Inverter	1	8.53
Total		60258.86

Tabla 5-6: Dimensión bloque del circuito total de 1 canal Invertido.

<b>Circuito Transmisor Total</b>		
<b>Bloque</b>	<b>Cantidad</b>	<b>Area (µm²)</b>
Circuito Total 1 Ch	1	60250.33
Circuito Total 1 Ch Inv	1	60258.86
Total		120509.19

Tabla 5-7: Dimensión bloque completo del Transmisor.

Con el área estimada por bloque (Tabla 5-1 a Tabla 5-7), en una primera aproximación, el acomodo de los bloques queda como se indica en la Figura 5-2.

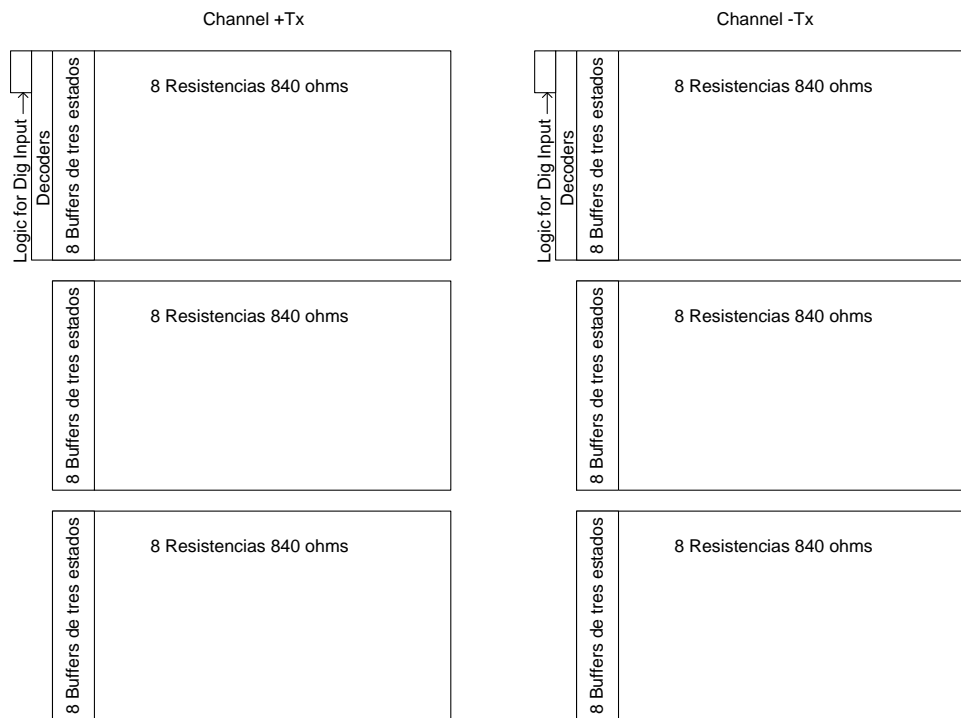


Figura 5-2: Layout preliminar del transmisor.

## 5.2 Diseño de layout de celdas básicas del módulo Transmisor

A continuación se presenta una primera aproximación del diseño del layout a la medida (custom) de los bloques que conforman el transmisor del módulo SerDes. Esta primera aproximación del layout servirá para una estimación del área de silicio que ocupará este módulo. Nótese que los bloques custom tienen una determinada dimensión vertical y una dimensión horizontal, que se modificarán de acuerdo con el acomodo del floorplan tanto para optimizar el área como para minimizar las parásitas del layout.

En el diseño del layout, se usaron los tamaños mínimos de transistores que permite la tecnología CMOS de 180 nm con el respectivo escalamiento para el pMOS según la relación de los transistores nMOS y pMOS (1:~2.5 a 1:5). Para algunas celdas analógicas, se usaron los tamaños mínimos de L y se escalaron los valores de W siguiendo la regla de escalamiento de la tecnología.

$$W_{layout} = 2x(\text{borde}) + nx(\text{tamaño-contacto}) + (n-1)x(\text{separación-contacto})$$

donde  $n$  es el número de contactos.

En el proceso CMRF7SF de IBM, la distancia del contacto al borde de la capa de difusión es de 100 nm; el tamaño de contactos es de 200 nm y la separación-contactos es de 270 nm. Así, la  $W_{\min}$  que se usó es de 400 nm y la  $L_{\min} = 200$  nm.

### 5.2.1 Layout del circuito Inversor:

La Figura 5-3 presenta el layout del circuito inversor, la Tabla 5-8 presenta las dimensiones del mismo:

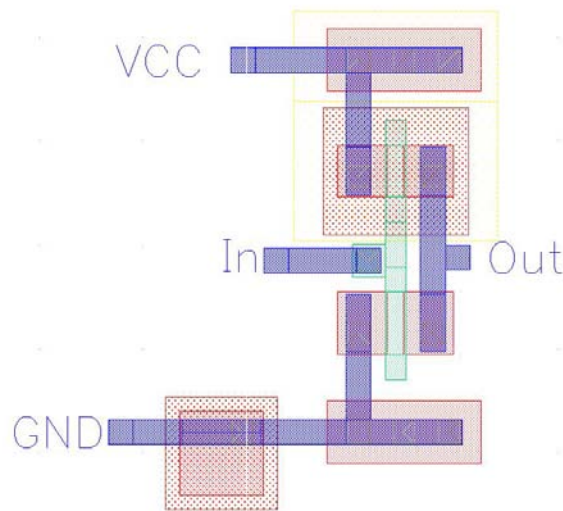


Figura 5-3: Layout para el inversor

Transistor	W	L
TP0	500nm	200nm
TN0	600nm	200nm

Tabla 5-8: Dimensiones circuito inversor

El layout de las celdas diseñadas se verificó con **Assura-Cadence**. Debido a que las celdas individuales sufren cambios principalmente en la región nWell cuando se realiza el acomodo de éstas al formar los bloques más grandes del módulo transmisor, se opta por no hacer la prueba de reglas de diseño (DRC) a nivel de celda básica sino cuando se integra a bloques más grandes. La prueba obligada para todas las celdas diseñadas es la de comparación entre las vistas layout y esquemático (prueba LVS).



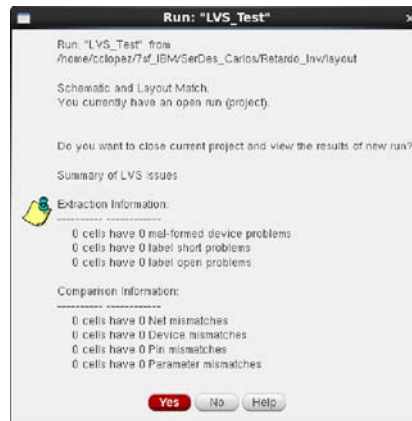


Figura 5-4: Resultados de la prueba **Assura**-LVS de la celda inversor.

Puede observarse de la Figura 5-4, que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda inversor.

### 5.2.2 Layout de compuerta NOR de dos entradas (NOR2)

La Figura 5-5 presenta el layout de la compuerta NOR de dos entradas y la Tabla 5-9 muestra sus dimensiones:

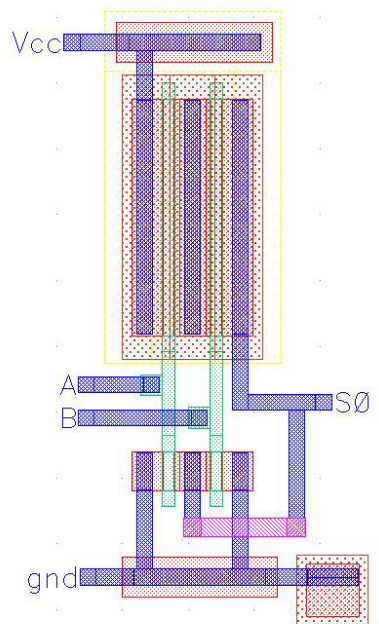


Figura 5-5: Layout compuerta NOR.

Transistor	W	L
TP0	3.6um	200nm
TP1	3.6um	200nm
TN0	600nm	200nm
TN1	600nm	200nm

Tabla 5-9: Dimensiones Compuerta NOR.

Estas compuertas se pueden reducir en tamaño dependiendo el acomodo que se requiera en el floorplan, su función es el control digital de las diferentes características del transmisor.

No hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda NOR de 2 entradas como ocurrió con la celda inversora Figura 5-4.

### 5.2.3 Layout de la compuerta NAND de dos entradas, (NAND2)

La Figura 5-6 presenta el layout de la compuerta NAND de dos entradas y la Tabla 5-10 sus dimensiones:

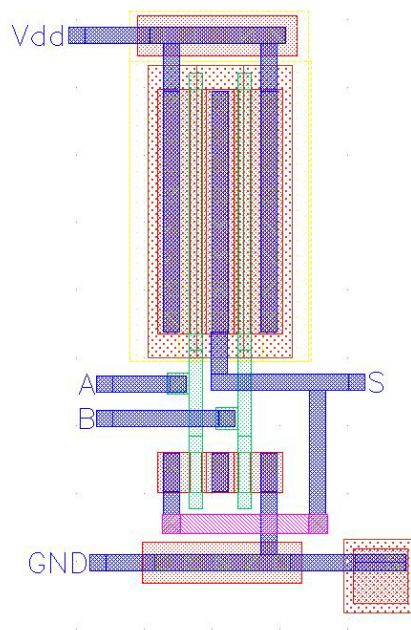


Figura 5-6: Compuerta NAND

Transistor	W	L
TP0	3.6um	200nm
TP1	3.6um	200nm
TN0	600nm	200nm
TN1	600nm	200nm

Tabla 5-10: Dimensiones Compuerta NAND

Estas compuertas se pueden reducir en tamaño dependiendo el acomodo que se requiera en el floor plan, su función es el control digital de las diferentes características del transmisor.

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda NAND de 2 entradas como ocurrió con la celda inversora Figura 5-4.

#### 5.2.4 Layout de la compuerta NOR de tres entradas (NOR3)

La Figura 5-7 presenta el layout de la compuerta NOR de tres entradas y la Tabla 5-11 muestra sus dimensiones:

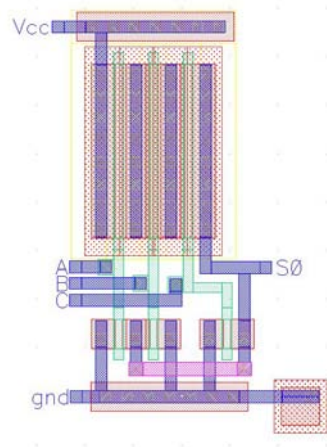


Figura 5-7: Compuerta NOR de tres entradas.

Transistor	W	L
TP0	3.6um	200nm
TP1	3.6um	200nm
TP2	3.6um	200nm
TN0	600nm	200nm
TN1	600nm	200nm
TN3	600nm	200nm

Tabla 5-11: Dimensiones Compuerta NOR3

Estas compuertas se pueden reducir en tamaño dependiendo el acomodo que se requiera en el floor plan, su función es el control digital de las diferentes características del transmisor.

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda NOR de 3 entradas como ocurrió con la celda inversora Figura 5-4.

### 5.2.5 Layout de la compuerta NAND de tres entradas (NAND3)

La Figura 5-8 presenta el layout de la compuerta NAND de tres entradas y la Tabla 5-12 muestra sus dimensiones:

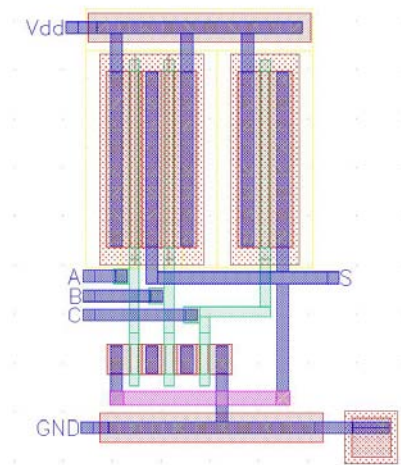


Figura 5-8: Compuerta NAND de tres entradas.

Transistor	W	L
TP0	3.6um	200nm
TP1	3.6um	200nm
TP3	3.6um	200nm
TN0	600nm	200nm
TN1	600nm	200nm
TN4	600nm	200nm

Tabla 5-12: Dimensiones Compuerta NAND3

Estas compuertas se podrán reducir en tamaño dependiendo el acomodo que se requiera en el floorplan, su función es el control digital de las diferentes características del transmisor.

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda NAND de 3 entradas como ocurrió con la celda inversora Figura 5-4.

### 5.2.6 Layout del circuito amarrador (tied).

La Figura 5-9 presenta el layout del circuito amarrador a valor Alto o valor Bajo y la Tabla 5-13 muestra sus dimensiones:

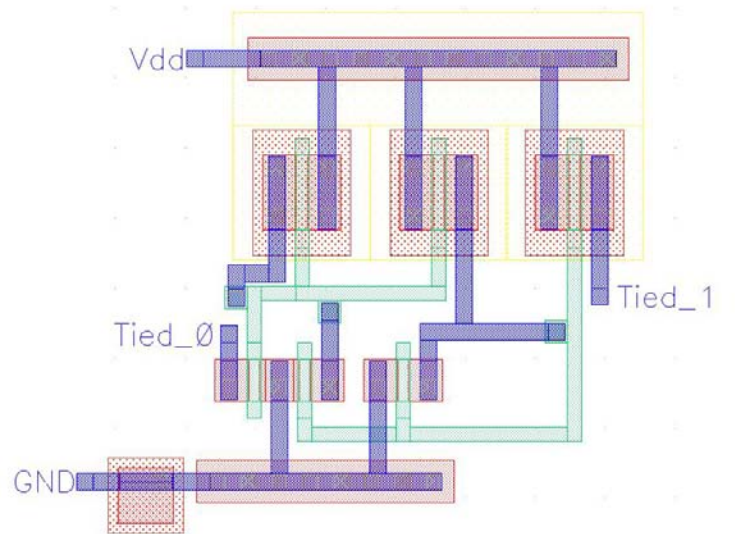


Figura 5-9: Circuito Amarrador (Tied)

Transistor	W	L
TP0	1.08um	200nm
TP1	1.08um	200nm
TP2	1.08um	200nm
TN0	600nm	200nm
TN1	600nm	200nm
TN2	600nm	200nm

Tabla 5-13: Dimensiones Circuito Tied

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas layout y schematic de la celda Tied como ocurrió con la celda inversora Figura 5-4.

### 5.2.7 Layout del Buffer de tres estados

El buffer de tres estados es el layout individual más grande del transmisor, y requiere de duplicarlo y añadirle dos resistencias para generar la celda básica de control de impedancia. La generación de esta celda básica impactara grandemente el floorplan de transmisor.

La Figura 5-10 presenta el layout del Buffer de tres estados y la Tabla 5-14 muestra sus dimensiones:

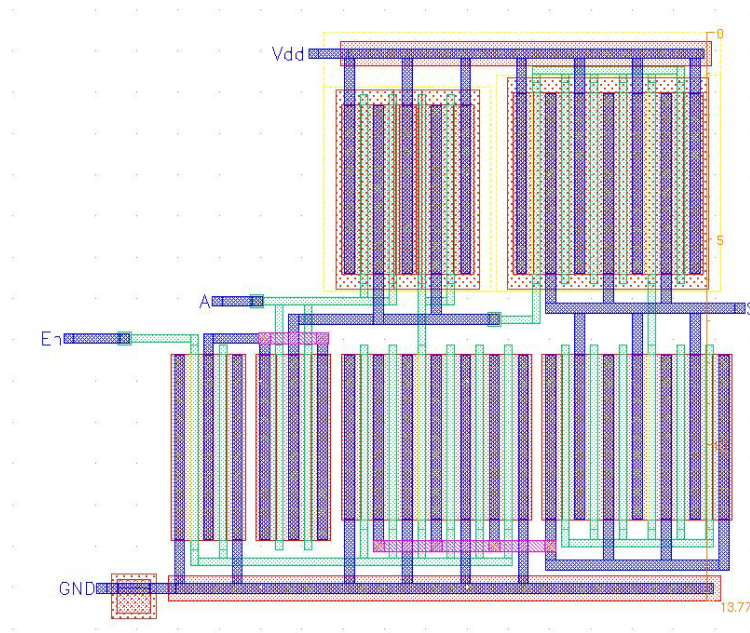


Figura 5-10: Buffer de tres estados.

Transistor	W	L
TP0	8.1um	200nm
TP1	8.1um	200nm
TP2	20.4um	200nm
TN0	9um	200nm
TN1	9um	200nm
TN2	23.8um	200nm
TN3	23.8um	200nm

Tabla 5-14: Dimensiones Buffer de tres estados

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda Buffer de tres estados como ocurrió con la celda inversora Figura 5-4. Se presentó una discrepancia entre las vistas *layout* y *schematic* del tipo *parameter* que está relacionado con el tamaño de los transistores. Este error LVS aparece cuando se usa el parámetro *Multifinger* o *Multiplicity* de las *pCells* de los transistores en la versión actual del proceso CMRF7SF de IBM. Este error LVS no es un error propiamente debida al diseño de layout de la celda sino a un error de configuración de las *pCells* de los transistores. El error se corrigió cambiando la función de división de transistores de *fingers* a *multiply* para evitar que la *pCell* hiciera un cálculo erróneo de las dimensiones W de los transistores del esquemático y del layout.



### 5.2.8 Layout del Inversor-Retardador

La Figura 5-11 presenta el layout de al circuito Retardador-Inversor y la Tabla 5-15 sus dimensiones:

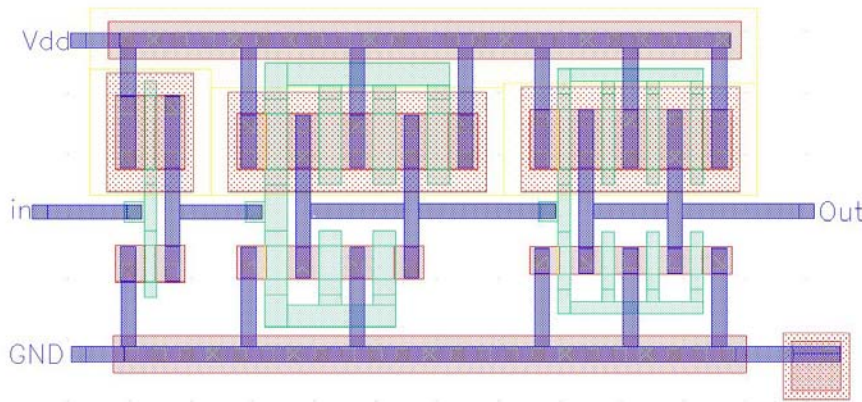


Figura 5-11: Buffer Retardador-Inversor

Transistor	W	L
TP0	680nm	360nm
TP1	3.6um	360nm
TP2	3.9um	200nm
TN0	400nm	360nm
TN1	1.62um	360nm
TN2	1.8um	200nm

Tabla 5-15: Dimensiones Retardador Inversor

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda Buffer de tres estados como ocurrió con la celda inversora Figura 5-4. Se presento una discrepancia entre las vistas *layout* y *schematic* del tipo *parameter* que está relacionado con el tamaño de los transistores.

Este error LVS no es un error propiamente debida al diseño de layout de la celda sino a un error de configuración de las *pCells* de los transistores. Se corrigió cambiando parámetros de los componentes tanto en layout como en esquemático, por tanto se considera que la celda pasa la prueba LVS.

### 5.2.9 Layout del Multiplexor

La Figura 5-12 presenta el layout del multiplexor 2 a 1 y la Tabla 5-16 muestra sus dimensiones:

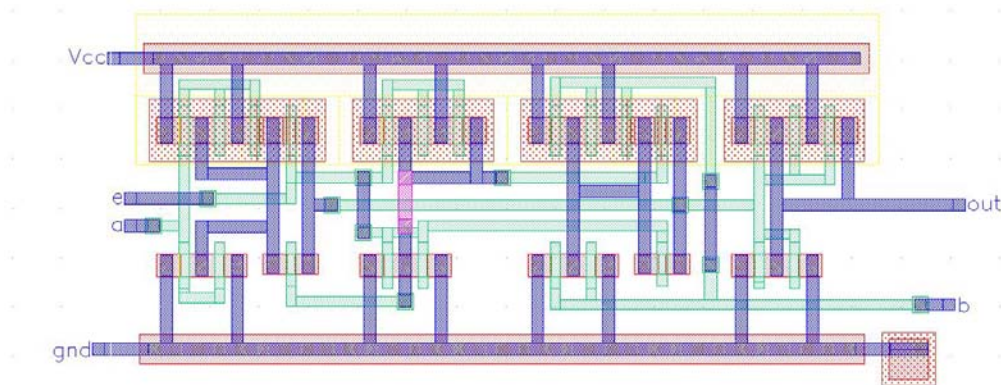


Figura 5-12: Multiplexor de 2 entradas

Transistor	W	L
TP0	1.62um	200nm
TP1	1.62um	200nm
TP2	1.62um	200nm
TP5	1.62um	200nm
TP7	540nm	200nm
TP8	540nm	200nm
TN0	900nm	200nm
TN1	900nm	200nm
TN2	900nm	200nm
TN5	900nm	200nm
TN6	400nm	200nm
TN7	400nm	200nm

Tabla 5-16: Dimensiones Circuito MUX

El multiplexor es el circuito que más transistores tiene aunque no con dimensiones grandes. Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda Buffer de tres estados como ocurrió con la celda inversora Figura 5-4. Se presentó una discrepancia entre las vistas *layout* y *schematic* del tipo *parameter* que está relacionado con el tamaño de los transistores.

Este error LVS no es un error propiamente debida al diseño de layout de la celda sino a un error de configuración de las *pCells* de los transistores. Se corrigió cambiando parámetros de los componentes tanto en layout como en esquemático, por tanto se considera que la celda pasa la prueba LVS.



### 5.2.10 Layout del circuito Buffer.

La Figura 5-13 presenta el layout del circuito Buffer y la Tabla 5-17 sus dimensiones:

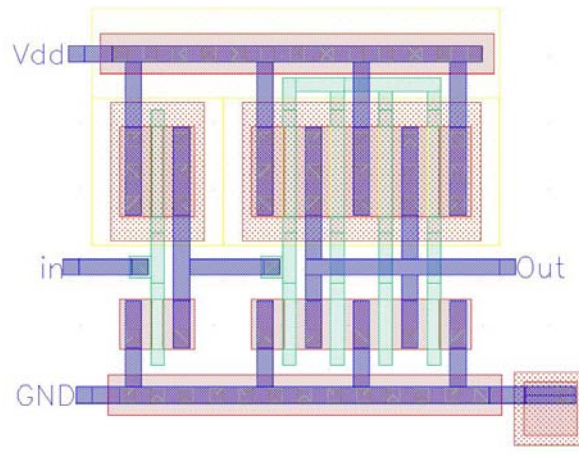


Figura 5-13: Layout del circuito Buffer

Transistor	W	L
TP0	1.35um	200nm
TP1	5.4um	200nm
TN0	750nm	200nm
TN1	3.0um	200nm

Tabla 5-17: Dimensiones Circuito Buffer

Se observó que no hay discrepancias de nodos, de puertos o de número de componentes en las vistas *layout* y *schematic* de la celda Buffer de tres estados como ocurrió con la celda inversora Figura 5-4. Se presentó una discrepancia entre las vistas *layout* y *schematic* del tipo *parameter* que está relacionado con el tamaño de los transistores.

Este error LVS no es un error propiamente debida al diseño de layout de la celda sino a un error de configuración de las *pCells* de los transistores. Se corrigió cambiando parámetros de los componentes tanto en layout como en esquemático, por tanto se considera que la celda pasa la prueba LVS.

### 5.3 Layout completo del módulo transmisor

En esta sección se desarrollará el circuito transmisor completo utilizando los bloques básicos presentados en la sección anterior. Este desarrollo corresponde a un canal de transmisión, como se explicó en la propuesta de este proyecto, el transmisor es diferencial por lo que requiere de dos canales, el segundo canal es la duplicación del desarrollo mostrado a continuación.

### 5.3.1 Buffer de Tres Estados

El bloque presentado en la Figura 5-14 es el Buffer de tres estados junto con dos resistencias de 820  $\Omega$ . Este bloque es el que nos da la posibilidad de modificar la impedancia de acople con la línea de transmisión.

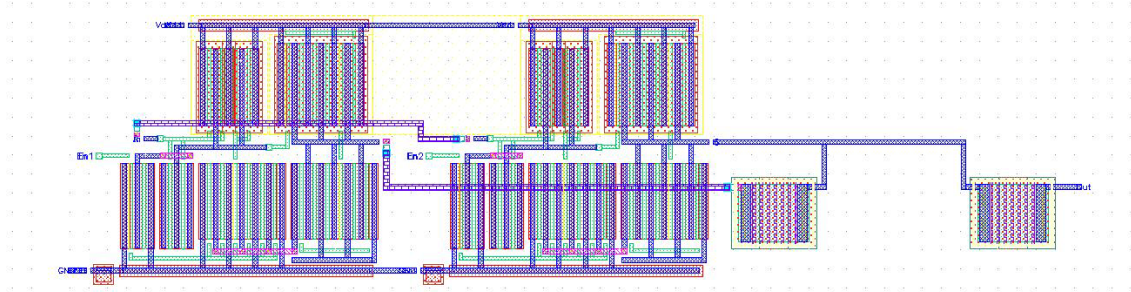


Figura 5-14: Buffer de tres estados con dos resistencias de 820  $\Omega$ .

### 5.3.2 Decodificador 2 Entradas 3 Salidas

A continuación se presenta en la Figura 5-15 el codificador de 2 entradas y 3 salidas tipo termómetro, este bloque se utiliza para el control de amplitud y pre-énfasis del transmisor:

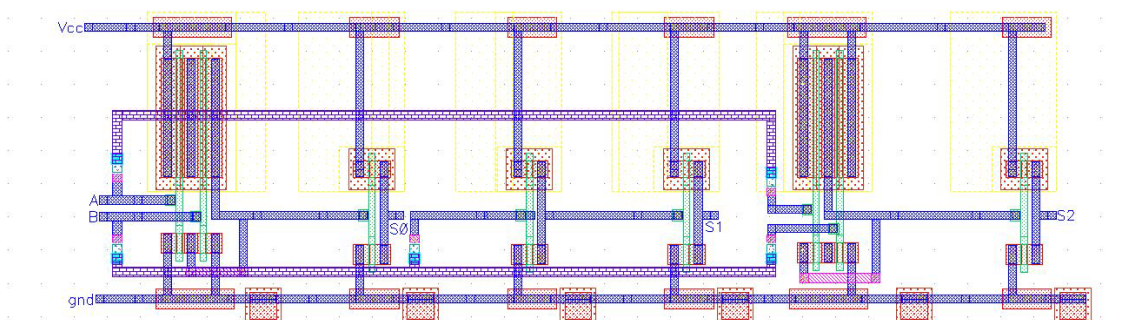


Figura 5-15: Codificador de 2 a 3 tipo termómetro



### 5.3.3 Decodificador 4 Entradas 12 Salidas

A continuación se presenta en la Figura 5-16 el codificador de 4 entradas y 12 salidas tipo termómetro, este codificador es el encargado de configurar la impedancia de acople con la línea de transmisión de datos:

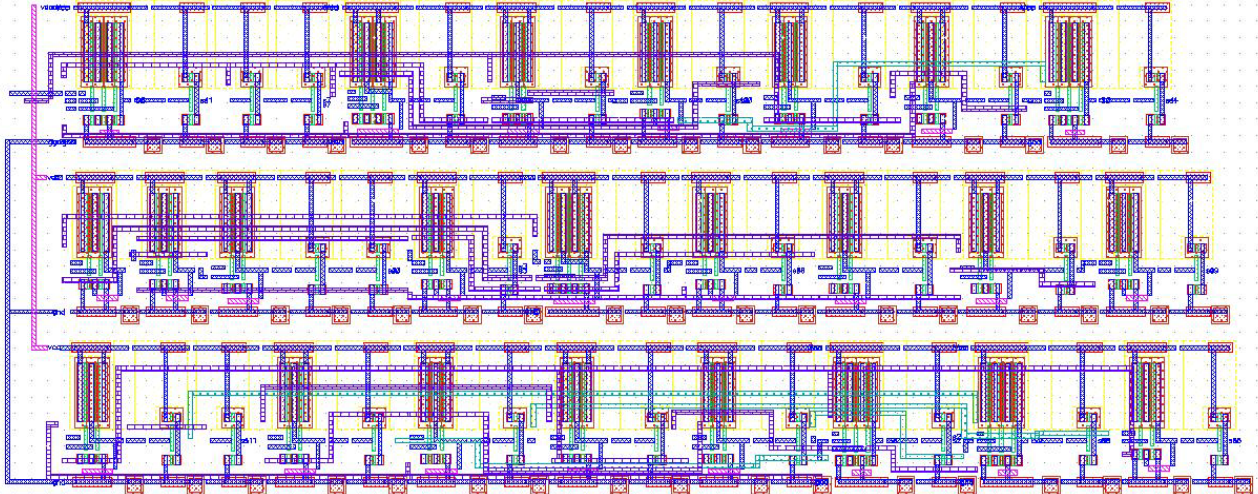


Figura 5-16: Codificador de 4 a 12 tipo termómetro.

### 5.3.4 Unidad ZAP

La unidad básica de control de impedancias, amplitud y pre énfasis conocida como bloque ZAP se muestra en la Figura 5-17, el transmisor completo está compuesto por tres de estas unidades de control:

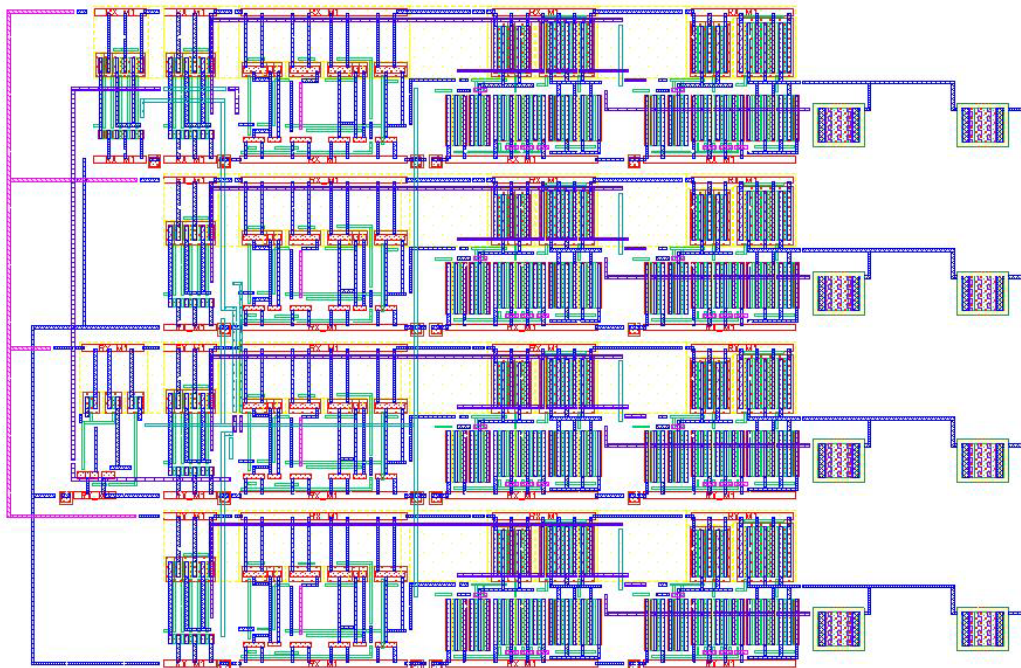


Figura 5-17: Unidad básica de control ZAP

### 5.3.5 Layout de un canal de módulo de Transmisión

El diseño total de un canal del transmisor se muestra a continuación en la Figura 5-18, esta es una primera aproximación al layout, las dimensiones de este bloque son de 170.98  $\mu\text{m}$  por 177.56  $\mu\text{m}$ , estas dimensiones difieren grandemente de la primera aproximación calculada en la sección 5.1 de este capítulo debido a las dimensiones de la resistencias, ya que el cálculo de la primera aproximación se hizo utilizando los parámetros de otra tecnología. Este diseño presenta áreas sin usar que se pueden rellenar con la circuitería del otro canal a transmitir (+TX, -TX) y con circuitería *Dummy* para mejorar la densidad del diseño.

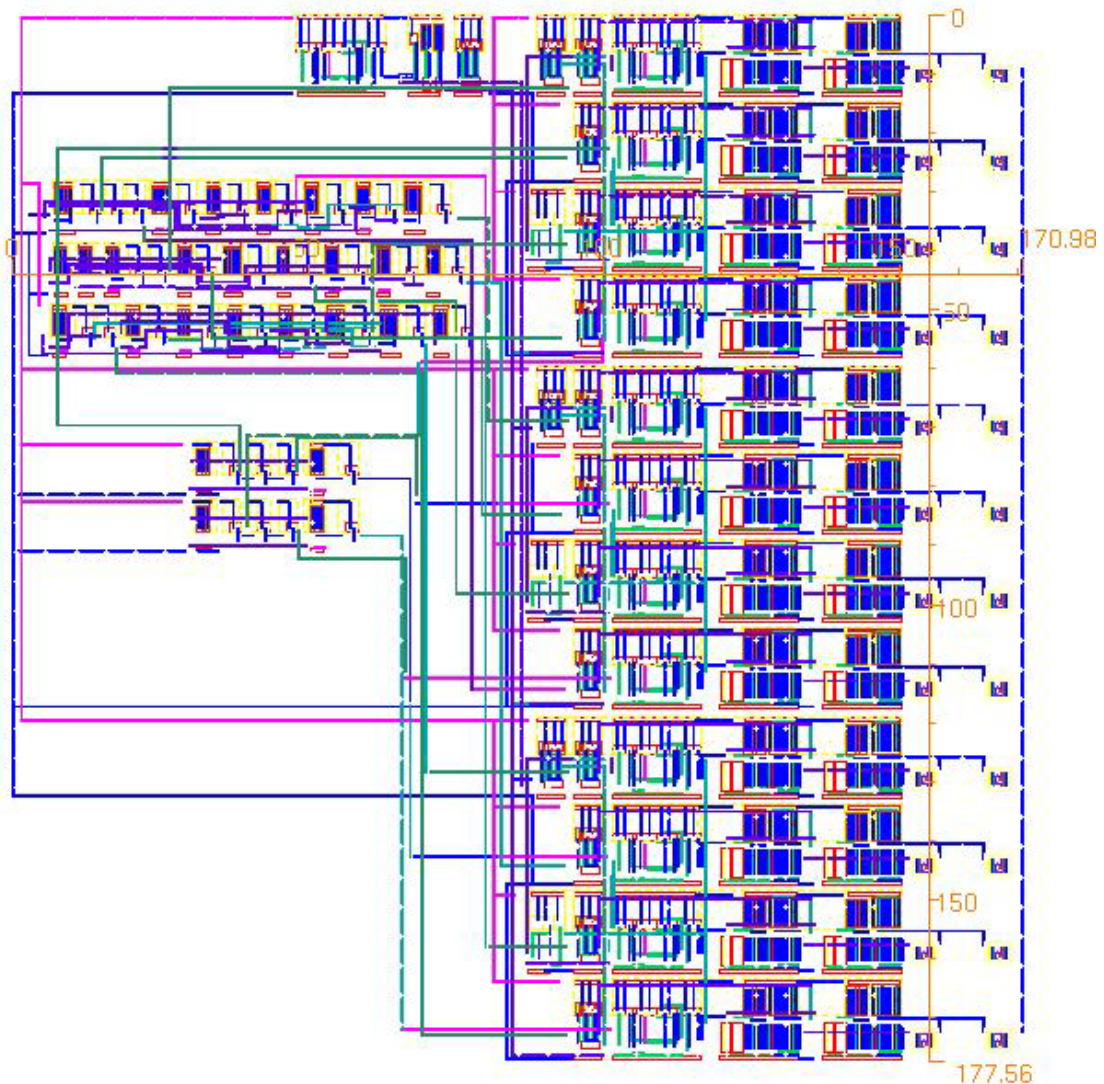


Figura 5-18: Diseño completo canal +TX

Se diseñó una primera versión el layout de las celdas individuales del módulo transmisor en modo *full custom*. La información obtenida de estas celdas sirvió para hacer una primera estimación, planeación y acomodo de los bloques (*floorplan*). El

layout de la Figura 5-18 corresponde a un canal de transmisión, puede ser cualquiera de ellos (+Tx y -Tx) no hay ninguna diferencia de parámetros de impedancia, máxima amplitud, etc. El segundo canal sería una duplicación del diseño de la Figura 5-18 en principio, pero puede ser espejado y optimizado en forma para mejorar el uso de área en el acomodo con el canal de complementario de transmisión.





## 6 Conclusiones

En este reporte presentamos el proceso de diseño de un transmisor diferencial serial para protocolo de comunicación PCI Express 1 que tiene la característica de poder controlar parámetros importantes para la optimización de la transmisión de datos como son la amplitud de la señal, pre-énfasis de la señal e impedancia de acople con la línea de transmisión, todo esto a través de una topología self-series terminated transmitter (SSSTT).

El diseño es basado en la conjunción de celdas básicas digitales, buffers y multiplexores las cuales fueron desarrolladas full custom por el autor de este trabajo, la herramienta **Cadence** presentó algunas limitantes debido a la configuración de las p-Cell de la prueba **Assura**-LVS ya que reportaba errores inexistentes en el diseño lo cual provocó retrasos en el desarrollo del proyecto, a pesar de que mucho de estas limitantes fueron resueltas, el tiempo invertido en ellas tuvo un impacto fuerte en el ya de por si limitado tiempo que se tenía para realizar el desarrollo y mandar construir el prototipo del proyecto SerDes, aun así, todas estas limitantes detectadas pueden servir para que futuras generaciones de estudiantes de la especialidad no las padezcan, o que para futuras implementaciones de nuevas tecnologías en el sistema **Cadence** tengan un check list de archivos y licencias requeridas para desarrollar proyectos completos.

El sistema **Cadence** es muy versátil y amigable de tal manera que el diseño full custom de las celdas fue un proceso lento pero bastante intuitivo en el manejo de la herramienta con ayudas que permitían minimizar las posibilidad de errores, además de las pruebas pre establecidas como el DRC y LVS que aseguran una mejor posibilidad de éxito con el diseño final y prototipado del sistema.

Los análisis de post layout se han dejado para una siguiente etapa en el diseño ya que en esta primera aproximación no se contó con el tiempo suficiente para hacerlas.

## 7 Lista de Referencias

- [1] Dave Lewis, “SerDes Architectures and Applications”, / National Semiconductor Corporation / 2004
  
- [2] Federico Lobato “Proyecto-G1”, /Private Communication / Spring 2015
  
- [3] Christian Weber, Jinjin He, Lizhi Charlie Zhong, and Huaping Liu “Multiband architecture for high-speed SerDes”, / This article is from a class at DesignCon 2011 / January 20, 2011
  
- [4] Ravi Budrunk, Don Anderson, Tom Shanley “PCI Express System Architecture”, /Addison-Wesley Developer’s Press / April 2008.
  
- [5] Romeo Covarrubias Larios y Enrique Maximiliano Calderon Ramos, “Diseño de un transmisor serial de datos con Impedancia, Amplitud y Pre Énfasis Programables”, /Instituto Tecnológico de Estudios Superiores de Occidente / Tlaquepaque, Jalisco, Feb 2008, 109 páginas.
  
- [6] Christian Menolfi, Thomas Toifl, Peter Buchmann, Marcel Kossel, Thomas Morf, Jonas Weiss, Martin Schmatz , “A 16Gb/s Source-Series Terminated Transmitter in 65nm CMOS SOI”, / IBM, Rueschlikon, Switzerland / 2007
  
- [7] Marcel Kossel, Christian Menolfi, Jonas Weiss, Peter Buchmann, George von Bueren, Lucio Rodoni, Thomas Morf, Thomas Toifl, Martin Schmatz, “A T-Coil-Enhanced 8.5Gb/s High-Swing Source-Series-Terminated Transmitter in 65nm Bulk CMOS”, / IBM Zurich Research Laboratory, Rüschrlikon, Switzerland / 2008
  
- [8] Esteban Martinez “Diseño de Circuito Inversor”, /Private Communication / Summer 2015
  
- [9] Federico Lobato “PWL voltage source code in Python for a random sequence of bits”, /Private Communication / Fall 2015



[10] Jan M Rabaey, Anantha P. Chandrakasan y Borivoje Nikolic "Digital Integrated Circuits. A Design Perspective, Second Edition", / Prentice Hall, / Estados Unidos 2002, 761 páginas.

[11] Rashed Zafar Bhatti, Monty Denneau, Jeff Draper "2 Gbps SerDes Design Based on IBM Cu-11 (130nm) Standard Cell Technology", / University of Southern California Marina del Rey, CA-90292, USA / 2008

## 8 Apéndice

### 8.1 Ecuación del Divisor de Voltaje

La Ecuación 8-1 muestra la fórmula para el divisor de voltaje de dos resistencias:

Ecuación 8-1 
$$V_{out} = \frac{R_A}{R_A + R_B} * V_{in}$$

De esta fórmula haciendo manejos algebraicos despejamos la resistencia  $R_B$  para poder hacer los cálculos de la impedancia que el transmisor serial está configurando para acoplarse con la línea de comunicaciones:

Ecuación 4-1 
$$R_B = \left[ \frac{V_{in}}{V_{out}} * R_A \right] - R_A$$