
This is the **published version** of the bachelor thesis:

Sánchez Miras, Francesc. *Implementación CMOS de un circuito de beamforming para transductores piezoeléctricos de ultrasonido*. Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Grau en Enginyeria Electrònica de Telecomunicació)

This version is available at <https://ddd.uab.cat/record/336120>

under the terms of the  license.



TREBALL DE FI DE GRAU

GRAU EN ENGINYERIA ELECTRÒNICA DE TELECOMUNICACIÓ

Implementación CMOS de un circuito de *beamforming* para transductores piezoeléctricos de ultrasonido

Francesc Sánchez Miras

NIU: 1666877

DIRECTORA: Arantxa Uranga

DEPARTAMENT D'ENGINYERIA ELECTRÒNICA

UNIVERSITAT AUTÒNOMA DE BARCELONA

Bellaterra, juny de 2026

Resumen y palabras clave

Este Trabajo Fin de Grado presenta la implementación CMOS de una arquitectura digital de *beamforming* para transductores piezoeléctricos de ultrasonido. El sistema se orienta al control temporal de un array de ocho canales mediante una cadena de retardos programables, con el objetivo de modificar la dirección efectiva del haz ultrasónico a partir de desfases discretos entre canales.

La memoria introduce primero la base física del problema, incluyendo ultrasonidos, PMUTs, arrays y la relación entre retardo temporal y ángulo de direccionamiento. A continuación se definen las especificaciones principales del sistema y se describe la arquitectura propuesta, formada por lógica de configuración, bloque de entrada, cadena de retardo, selección de sentido y etapas de adaptación de nivel. Posteriormente se desarrolla la implementación a nivel de transistor de las celdas digitales principales y se estudia su comportamiento mediante simulaciones DC y temporales. Finalmente, se presenta la implementación física mediante *layout* en tecnología CMOS HV de SilTerra de 0.13 μm , junto con una validación funcional a nivel de sistema y las conclusiones generales del trabajo.

Palabras clave: CMOS HV, *beamforming*, ultrasonidos, PMUT, retardos programables, SilTerra, Cadence Virtuoso.

Abstract

This Bachelor's Thesis presents the CMOS implementation of a digital *beamforming* architecture for piezoelectric ultrasonic transducers. The system is focused on the timing control of an eight-channel array by means of a programmable delay chain, with the aim of modifying the effective direction of the ultrasonic beam through discrete delays between channels.

The document first introduces the physical background of the problem, including ultrasound, PMUTs, arrays and the relationship between time delay and steering angle. The main system specifications are then defined, followed by the proposed architecture, which includes configuration logic, input control, delay chain, direction selection and level-shifting stages. The transistor-level implementation of the main digital cells is subsequently developed and analysed through DC and temporal simulations. Finally, the physical layout implementation in SilTerra 0.13 μm CMOS HV technology is presented, together with a system-level functional validation and the overall conclusions of the work.

Keywords: CMOS HV, *beamforming*, ultrasound, PMUT, programmable delays, SilTerra, Cadence Virtuoso.

Agradecimientos

Quiero agradecer en primer lugar a mi directora, Arantxa Uranga, por su orientación, seguimiento y ayuda durante el desarrollo de este Trabajo Fin de Grado. Sus comentarios han sido fundamentales para aprender, ordenar el trabajo, mejorar la memoria y enfocar correctamente la validación del sistema e implementación física.

También quiero agradecer al Departamento de Ingeniería Electrónica conjuntamente al grupo ECAS (Electronic Circuits and Systems Research Group) y al entorno de trabajo del proyecto por facilitar los recursos y herramientas necesarios para el desarrollo del diseño, las simulaciones y la implementación física.

Por último, quiero dar las gracias a mi familia, amigos y compañeros por el apoyo recibido durante todo el proceso. Este trabajo no solo ha supuesto el desarrollo de una arquitectura electrónica concreta, sino también una etapa de aprendizaje técnico y personal muy importante, no hubiese sido posible sin su apoyo a lo largo del grado.

Índice general

Resumen y palabras clave	i
Abstract	ii
Agradecimientos	iii
Lista de figuras	viii
Lista de tablas	xi
1. Introducción	1
1.1. Motivación y objetivos	1
1.2. MEMS: sistemas microelectromecánicos	2
1.3. Principio de piezoelectricidad	2
1.4. Ultrasonido y sistemas ultrasónicos básicos	3
1.4.1. Funcionamiento general	4
1.4.2. Método pulso-eco	4
1.4.3. Time of Flight	4
1.4.4. Bloques básicos de un sistema ultrasónico	5
1.5. Transductores ultrasónicos y PMUTs	5
1.5.1. Qué es un PMUT	6
1.5.2. Propiedades importantes de los PMUTs	6
1.6. Arrays, <i>beamforming</i> y retardos temporales	6
1.6.1. Concepto de <i>beamforming</i>	7
1.6.2. Separación entre elementos	7

1.6.3. Retardo entre elementos adyacentes	8
1.6.4. Relación con una cadena digital de retardos	9
1.7. Contexto del proyecto	10
1.8. Organización del proyecto	10
2. Especificaciones y planteamiento del sistema	11
2.1. Especificaciones generales del sistema	12
2.2. Frecuencia de reloj y pasos temporales disponibles	12
2.2.1. Pasos temporales disponibles y número de posiciones	14
2.2.2. Frecuencia de reloj requerida para distintos PMUTs	15
2.3. Barrido angular para un PMUT de 5 MHz	15
2.4. Gráficas de barrido para distintas frecuencias de PMUT	17
3. Arquitectura general propuesta	19
3.1. Vista general de la arquitectura	20
3.2. Codificación del retardo y selección de sentido	20
3.3. Fase de programación y fase de transmisión	22
3.3.1. Bloque de configuración	23
3.3.2. Bloque de transmisión	24
3.4. Distribución conceptual de los ocho canales	25
3.5. Implementación circuital del canal y circuitería de selección de retardo	27
3.5.1. Cadena DTC y selección del retardo programado	27
4. Implementación a nivel de transistor	30
4.1. Inversor CMOS	30
4.2. Buffer CMOS	32
4.3. Puerta NAND CMOS de dos entradas	35
4.4. Puerta NOR CMOS de dos entradas	37
4.5. Puerta NAND CMOS de tres entradas	40
4.6. Puerta XOR CMOS	43
4.7. Flip-flop tipo D con reset asíncrono activo bajo	46

4.7.1. Respuesta temporal del flip-flop	48
4.8. Level shifter de 1.5 V a 5 V	48
4.9. Resumen de parámetros temporales extraídos	50
4.10. Criterio de dimensionamiento	50
4.11. Resumen	51
5. Implementación física mediante layout	53
5.1. Proceso general de diseño físico	54
5.1.1. Tecnología CMOS HV SilTerra de 130 nm	54
5.1.2. Compactación de celdas, pozos y anillos de guarda	55
5.1.3. Criterios generales de layout	55
5.1.4. Dimensiones físicas de los bloques	56
5.2. Layouts de bloques y celdas implementadas	57
5.2.1. Layout del buffer CMOS	57
5.2.2. Layout del flip-flop tipo D	58
5.2.3. Layout del bloque IN	58
5.2.4. Layout del registro serie-paralelo	59
5.2.5. Layout del DTC	60
5.2.6. Layout de la lógica de selección del DTC	61
5.2.7. Layout del multiplexor 2:1	62
5.2.8. Layout del multiplexor 8:1	63
5.2.9. Layout del level shifter de 1.5 V a 5 V	64
5.3. Integración física de bloques	65
5.3.1. Layout de un canal completo	66
5.3.2. Layout del sistema completo de ocho canales	67
5.4. Resumen	68
6. Validación funcional a nivel de sistema	69
6.1. Secuencia funcional validada	70
6.2. Validación del retardo programable	70
6.3. Validación multicanal y selección de sentido	72

6.4. Simulación transitoria de las salidas multicanal	73
6.5. Relación con el ángulo de barrido	76
6.6. Representación funcional de la adaptación de nivel	76
6.7. Resumen de validaciones funcionales	77
6.8. Alcance de la validación funcional	78
7. Conclusiones	79
Bibliografía	81

Lista de figuras

1.1. Representación conceptual de los efectos piezoeléctricos en un PMUT.	3
1.2. Relación geométrica entre la separación de los elementos del array, la diferencia de camino y el ángulo de direccionamiento del frente de onda. En la figura, d representa la separación entre elementos, equivalente a p en la notación utilizada en este trabajo. [6]	8
2.1. Ángulo de barrido en función de f_{clk} para un PMUT de 1 MHz.	17
2.2. Ángulo de barrido en función de f_{clk} para un PMUT de 2 MHz.	17
2.3. Ángulo de barrido en función de f_{clk} para un PMUT de 5 MHz.	17
2.4. Ángulo de barrido en función de f_{clk} para un PMUT de 10 MHz.	17
2.5. Ángulo de barrido en función de f_{clk} para un PMUT de 20 MHz.	18
3.1. Vista general de la arquitectura propuesta a nivel de bloques principales.	20
3.2. Separación temporal entre la fase de programación y la fase de transmisión. (Los pulsos a transmitir se representan de forma conceptual y no están dibujados a escala temporal real)	22
3.3. Esquema conceptual del bloque de configuración.	24
3.4. Esquema conceptual del bloque de transmisión.	25
3.5. Distribución conceptual de los ocho canales dentro del bloque de transmisión.	26
3.6. Esquema funcional interno de un canal de transmisión.	27
3.7. Estructura interna del DTC para la generación y selección del retardo programado.	28
4.1. Implementación CMOS de un inversor.	31
4.2. Simulación DC del inversor CMOS.	31
4.3. Respuesta temporal del inversor CMOS.	32
4.4. Implementación CMOS de un buffer mediante dos inversores en cascada.	33

4.5. Simulación DC del buffer CMOS.	34
4.6. Respuesta temporal del buffer CMOS.	35
4.7. Implementación CMOS de una puerta NAND de dos entradas.	36
4.8. Simulación DC de la puerta NAND CMOS de dos entradas.	36
4.9. Respuesta temporal de la puerta NAND CMOS de dos entradas.	37
4.10. Implementación CMOS de una puerta NOR de dos entradas.	38
4.11. Simulación DC de la puerta NOR CMOS de dos entradas.	39
4.12. Respuesta temporal de la puerta NOR CMOS de dos entradas.	40
4.13. Implementación CMOS de una puerta NAND de tres entradas.	41
4.14. Simulación DC de la puerta NAND CMOS de tres entradas.	42
4.15. Respuesta temporal de la puerta NAND CMOS de tres entradas.	42
4.16. Implementación CMOS de una puerta XOR de dos entradas.	43
4.17. Simulación DC de la puerta XOR CMOS.	44
4.18. Respuesta temporal de la puerta XOR CMOS.	45
4.19. Implementación lógica del flip-flop tipo D utilizado en la cadena de retardo. . .	46
4.20. Respuesta temporal del flip-flop tipo D considerando $C_L = 100$ fF y Reset=1 . .	48
4.21. Implementación transistor-level del <i>level shifter</i> de 1.5 V a 5 V.	49
5.1. Layout del buffer CMOS implementado mediante dos inversores en cascada. . .	57
5.2. Layout del flip-flop tipo D con reset asíncrono activo bajo.	58
5.3. Layout del bloque IN, encargado de la habilitación de señales de entrada y reloj.	59
5.4. Layout del registro serie-paralelo de configuración.	60
5.5. Layout de la cadena principal del DTC.	60
5.6. Layout de la lógica de selección del DTC.	61
5.7. Layout del multiplexor 2:1.	62
5.8. Layout del multiplexor 8:1 utilizado para la selección del tap retardado.	63
5.9. Layout del level shifter de 1.5 V a 5 V.	64
5.10. Layout de un canal completo de transmisión.	66
5.11. Layout del sistema completo de ocho canales.	67
6.1. Validación funcional general del retardo introducido por el DTC.	71

6.2. Validación funcional del orden de activación de los ocho canales para ambos sentidos de propagación.	72
6.3. Simulación transitoria del sistema completo en dirección directa. La configuración aplicada mediante DATA_IN es 01100000, con $D7 = 0$, $f_{clk} = 100$ MHz, retardo de $2T_{clk} = 20$ ns entre canales y tres pulsos de transmisión a 20 MHz iniciados alrededor de 110 ns.	74
6.4. Simulación transitoria del sistema completo en dirección inversa. La configuración aplicada mediante DATA_IN es 11111000, con $D7 = 1$, $f_{clk} = 100$ MHz, retardo de $4T_{clk} = 40$ ns entre canales y tres pulsos de transmisión a 20 MHz iniciados alrededor de 110 ns.	75
6.5. Representación funcional de la adaptación de nivel desde la lógica digital hasta la salida de alta tensión.	77

Lista de tablas

2.1. Especificaciones generales del sistema.	12
2.2. Frecuencia de reloj requerida utilizando $n = 7$	15
2.3. Barrido angular completo para $f_{\text{PMUT}} = 5 \text{ MHz}$ y $f_{\text{clk}} = 100 \text{ MHz}$	16
3.1. Programación funcional del retardo mediante D0–D6.	21
3.2. Función del bit D7.	22
3.3. Codificación acumulativa del retardo vista desde el DTC.	29
4.1. Dimensiones de los transistores del inversor CMOS.	30
4.2. Tabla de verdad del inversor CMOS.	32
4.3. Dimensiones de los transistores del buffer CMOS.	33
4.4. Tabla de verdad del buffer CMOS.	34
4.5. Dimensiones de los transistores de la puerta NAND CMOS de dos entradas. . .	36
4.6. Tabla de verdad de la puerta NAND de dos entradas.	37
4.7. Dimensiones de los transistores de la puerta NOR CMOS de dos entradas. . . .	38
4.8. Tabla de verdad de la puerta NOR de dos entradas.	39
4.9. Dimensiones de los transistores de la puerta NAND CMOS de tres entradas. . .	40
4.10. Tabla de verdad de la puerta NAND de tres entradas.	41
4.11. Dimensiones de los transistores de la puerta XOR CMOS.	44
4.12. Tabla de verdad de la puerta XOR de dos entradas.	45
4.13. Dimensiones de los transistores del <i>level shifter</i> de 1.5 V a 5 V.	50
4.14. Parámetros temporales obtenidos mediante simulación temporal con $C_L = 100 \text{ fF}$. .	50
5.1. Dimensiones físicas de los principales bloques implementados.	56

6.1. Retardos programables del DTC para $f_{clk} = 100$ MHz.	71
6.2. Instantes de activación esperados en función del sentido de propagación.	72
6.3. Resumen de validaciones funcionales a nivel de sistema.	78

Capítulo 1

Introducción

Los ultrasonidos forman parte de una tecnología ampliamente utilizada en la actualidad. Están presentes en equipos médicos, sensores de distancia, sistemas industriales, dispositivos de consumo y muchas otras aplicaciones en las que se desea obtener información de un objeto o de un medio sin necesidad de contacto directo. Su interés principal reside en que permiten explorar el entorno mediante ondas mecánicas y extraer información a partir de su propagación, reflexión o atenuación.

Aunque el principio general puede parecer sencillo, un sistema ultrasónico real combina distintos conceptos físicos y tecnológicos. Para entender el trabajo desarrollado no basta con considerar únicamente la existencia de un transductor que emite ultrasonidos. También es necesario conocer el papel de los sistemas MEMS, el principio de piezoelectricidad, el funcionamiento básico de un sistema ultrasónico y la necesidad de controlar retardos temporales cuando se utilizan varios canales de emisión.

El objetivo de este capítulo es presentar esta base teórica de forma progresiva. En primer lugar se introduce la motivación del trabajo. A continuación se describen los sistemas MEMS, la piezoelectricidad, los ultrasonidos y los transductores ultrasónicos, prestando especial atención a los PMUTs. Finalmente, se explica el uso de arrays de transductores, el concepto de *beamforming* y la relación entre los retardos temporales y el direccionamiento del haz acústico.

1.1. Motivación y objetivos

La motivación de este trabajo nace de un problema concreto: cuando se desea controlar electrónicamente la dirección de un haz ultrasónico mediante varios canales, es necesario definir con precisión cuándo debe activarse cada uno de ellos. Si todos los canales emiten simultáneamente, el sistema presenta un comportamiento determinado; en cambio, si se introducen pequeños desfases temporales entre ellos, el haz puede orientarse hacia una dirección concreta.

Por tanto, el tiempo relativo entre canales deja de ser un detalle secundario y pasa a convertirse en una magnitud fundamental. Este control temporal es la base de muchas técnicas de *beamforming*,

y obliga a diseñar circuitería capaz de generar retardos conocidos, repetibles y programables.

En este contexto, el objetivo general del trabajo es estudiar una arquitectura digital orientada a la generación de retardos programables para aplicaciones de *beamforming* ultrasónico. Antes de entrar en la implementación concreta del sistema, resulta necesario justificar por qué dichos retardos son necesarios y qué función cumplen dentro de una arquitectura ultrasónica multicanal.

1.2. MEMS: sistemas microelectromecánicos

Los sistemas microelectromecánicos, conocidos habitualmente como MEMS por sus siglas en inglés (*MicroElectroMechanical Systems*), son dispositivos de tamaño reducido que integran partes mecánicas y eléctricas dentro de una misma tecnología de fabricación [1, 2]. A diferencia de un circuito puramente electrónico, un MEMS incorpora también estructuras físicas capaces de moverse, deformarse o vibrar.

La importancia de esta tecnología reside en que permite fabricar dispositivos muy pequeños, con buena repetibilidad y potencialmente integrables con tecnología CMOS. Gracias a ello, es posible construir sensores y actuadores compactos, de bajo consumo y adecuados para una gran variedad de aplicaciones.

Actualmente, los MEMS se emplean en acelerómetros, giroscopios, sensores de presión, micrófonos, actuadores ópticos y otros sistemas. En todos estos casos existe una idea común: una estructura mecánica responde a una excitación o a un fenómeno físico del entorno, y esa respuesta puede detectarse o aprovecharse eléctricamente.

En el campo de los ultrasonidos, la tecnología MEMS ha permitido desarrollar transductores más pequeños y integrables con la electrónica. Esto ha favorecido la aparición de arquitecturas más compactas y de matrices con un elevado número de elementos. Los PMUTs, dispositivos con los que se relaciona este trabajo, pertenecen precisamente a esta familia de sistemas microelectromecánicos.

1.3. Principio de piezoelectricidad

La piezoelectricidad es una propiedad física de ciertos materiales que permite convertir energía mecánica en energía eléctrica y, también, realizar el proceso inverso. Esta característica resulta especialmente útil en ultrasonidos, ya que permite que un mismo tipo de material pueda emplearse tanto para generar vibraciones como para detectar perturbaciones mecánicas.

Cuando un material piezoeléctrico se somete a una presión o deformación mecánica, aparece una redistribución de cargas en su interior y se genera una señal eléctrica en sus terminales. Este fenómeno se conoce como **efecto piezoeléctrico directo**. En sentido contrario, si se aplica un campo eléctrico sobre el material, este experimenta una deformación mecánica. Este segundo caso recibe el nombre de **efecto piezoeléctrico inverso**.

La existencia de estos dos efectos convierte a la piezoelectricidad en una herramienta fundamental para los sistemas ultrasónicos. Para emitir una onda acústica, se aplica una señal eléctrica al material piezoeléctrico y se fuerza su vibración. Para recibir una onda, la presión acústica deforma el material y produce una señal eléctrica que posteriormente puede amplificarse y procesarse.

De forma sencilla, la piezoelectricidad actúa como un puente entre el dominio eléctrico y el dominio mecánico. Gracias a ella, un transductor puede funcionar como emisor y como receptor.

La Figura 1.1 muestra esta naturaleza bidireccional en el caso de un PMUT. A la izquierda se representa la situación en la que una perturbación mecánica genera una señal eléctrica, mientras que a la derecha se muestra el caso inverso, en el que una excitación eléctrica da lugar a la emisión de ultrasonidos.

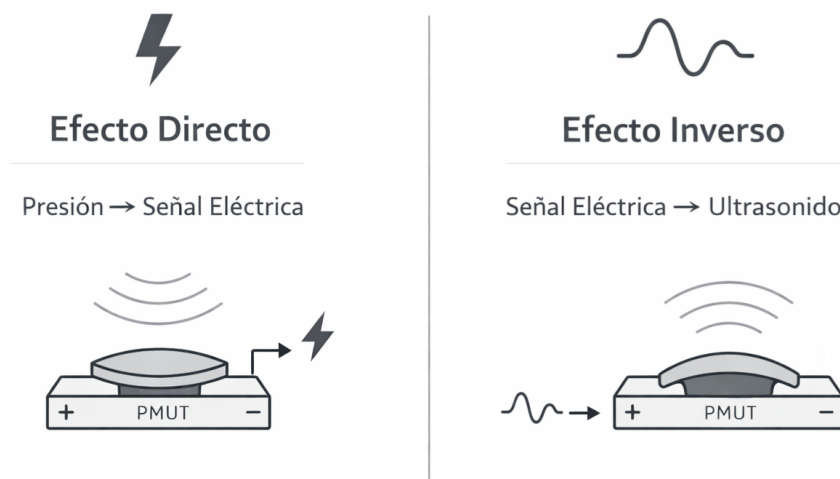


Figura 1.1: Representación conceptual de los efectos piezoeléctricos en un PMUT.

1.4. Ultrasonido y sistemas ultrasónicos básicos

Se denomina **ultrasonido** a toda onda acústica cuya frecuencia se encuentra por encima del límite superior de audición del ser humano, situado aproximadamente en 20 kHz. En muchas aplicaciones tecnológicas se trabaja a frecuencias bastante superiores, normalmente en el rango de los cientos de kHz o de los MHz.

A diferencia de las ondas electromagnéticas, las ondas ultrasónicas necesitan un medio material para propagarse. Pueden transmitirse a través de aire, agua, tejidos biológicos o sólidos, pero no a través del vacío. Además, su velocidad de propagación depende de las propiedades del medio. En este trabajo, dicha velocidad se representará mediante la letra c , que corresponde a la **velocidad de propagación del sonido en el medio**.

1.4.1. Funcionamiento general

Un sistema ultrasónico básico puede entenderse como un sistema que emite una onda acústica y analiza qué ocurre con ella después. En muchas aplicaciones, la onda se dirige hacia una región de interés, se refleja al encontrar un objeto o una discontinuidad, y una parte de esa energía regresa al sistema en forma de eco.

Ese eco contiene información útil. Si se conoce el instante de emisión y el instante de recepción, puede estimarse la distancia recorrida por la onda. Además, el análisis de la amplitud, la fase o la forma temporal del eco permite obtener información adicional sobre el objeto o sobre el medio en el que se ha producido la reflexión.

1.4.2. Método pulso-eco

Uno de los principios más utilizados en sistemas de ultrasonido es el llamado **método pulso-eco**. Su funcionamiento puede resumirse en los siguientes pasos:

1. Un transductor recibe una excitación eléctrica.
2. El transductor la transforma en una onda acústica.
3. La onda se propaga por el medio.
4. Al encontrar un objeto o una interfaz, una parte de la energía se refleja.
5. El eco reflejado regresa al transductor.
6. El transductor transforma ese eco en una señal eléctrica.

Este método resulta útil porque permite explorar un medio sin necesidad de contacto directo con el objeto de interés. Es la base de numerosos sistemas de medida de distancia, detección e imagen ultrasónica.

1.4.3. Time of Flight

A partir del método pulso-eco puede introducirse el concepto de **Time of Flight**, o tiempo de vuelo. El *Time of Flight* es el tiempo que transcurre entre el instante en que se emite la onda ultrasónica y el instante en que se recibe el eco reflejado.

Si se conoce la velocidad del sonido en el medio, el tiempo de vuelo permite calcular la distancia hasta el reflector. La relación más simple viene dada por:

$$d = \frac{ct_{\text{ToF}}}{2}$$

donde d es la distancia al objeto, t_{ToF} es el *Time of Flight* y c es la velocidad de propagación del sonido en el medio. El factor 2 aparece porque la onda recorre dos veces la distancia: primero viaja hasta el objeto y después regresa en forma de eco.

Por tanto, el *Time of Flight* permite relacionar tiempo y distancia dentro de un sistema ultrasónico. Esta idea se utiliza tanto en sensores de rango como en sistemas de imagen.

1.4.4. Bloques básicos de un sistema ultrasónico

Aunque existen muchas variantes, un sistema ultrasónico suele incluir los siguientes bloques:

- Un transductor o conjunto de transductores.
- Una etapa de transmisión, encargada de excitar el transductor.
- Una etapa de recepción, encargada de amplificar la señal recibida.
- Una lógica de control temporal.
- Una etapa de procesado, que interpreta la información contenida en los ecos.

En sistemas sencillos puede bastar con un único transductor. Sin embargo, cuando se desea dirigir el haz o controlar mejor la energía acústica en el espacio, se recurre a varios elementos funcionando de forma coordinada. En ese punto aparecen las matrices de transductores, los retardos temporales y las técnicas de *beamforming*.

1.5. Transductores ultrasónicos y PMUTs

Un **transductor ultrasónico** es el elemento encargado de transformar una señal eléctrica en una vibración mecánica y, en sentido contrario, convertir una vibración mecánica en una señal eléctrica. Es, por tanto, la pieza que conecta la electrónica con el fenómeno acústico.

Existen distintas tecnologías para implementar transductores ultrasónicos. Los dispositivos piezoeléctricos convencionales son una solución ampliamente utilizada. Sin embargo, cuando se busca miniaturización, integración y compatibilidad con procesos microelectrónicos, adquieren especial interés los transductores ultrasónicos micromecanizados.

Dentro de esta familia se encuentran los **PMUTs** (*Piezoelectric Micromachined Ultrasonic Transducers*), basados en el efecto piezoeléctrico y fabricados mediante tecnologías MEMS [2, 3, 4].

1.5.1. Qué es un PMUT

Un PMUT es un dispositivo MEMS formado, de manera general, por una membrana micromecanizada, una capa piezoeléctrica y un conjunto de electrodos.

Cuando se aplica una señal eléctrica a los electrodos, la capa piezoeléctrica se deforma. Esta deformación hace flexionar la membrana y genera una vibración mecánica. Si dicha vibración se transmite al medio, se obtiene una onda ultrasónica. En este caso, el PMUT trabaja como emisor.

Si el proceso ocurre en sentido contrario y una onda ultrasónica incide sobre la membrana, la deformación mecánica generada produce una señal eléctrica gracias al efecto piezoeléctrico directo. En este caso, el PMUT funciona como receptor.

1.5.2. Propiedades importantes de los PMUTs

A la hora de caracterizar un PMUT, existen varias propiedades especialmente relevantes:

- **Frecuencia de resonancia:** frecuencia a la que el dispositivo vibra con mayor eficiencia.
- **Ancho de banda:** rango de frecuencias en el que el dispositivo puede operar de forma útil.
- **Sensibilidad:** capacidad del transductor para convertir energía eléctrica en presión acústica durante la transmisión, normalmente expresada en Pa/V, y para convertir presión acústica en señal eléctrica durante la recepción, expresada en V/Pa.
- **Directividad:** variación de la respuesta del dispositivo en función de la dirección.
- **Integrabilidad:** facilidad para combinar el dispositivo con electrónica CMOS.

En este trabajo, la frecuencia de operación de los PMUTs se considera como un parámetro de diseño, ya que dependerá del dispositivo concreto utilizado. Por este motivo, en las expresiones teóricas se utilizará la variable f para representar de forma general la frecuencia de operación.

1.6. Arrays, *beamforming* y retardos temporales

Hasta ahora se ha considerado el funcionamiento de transductores individuales. Sin embargo, un único elemento ofrece una capacidad limitada para controlar la dirección del haz acústico y proporciona poca presión. Para obtener un mayor control espacial y mayor presión se utilizan conjuntos de transductores organizados en forma de matriz o *array*.

Un **array** de transductores es un conjunto de elementos colocados de manera ordenada y suficientemente próximos entre sí. La ventaja de esta configuración es que cada elemento puede emitir en un instante ligeramente distinto. Esa diferencia temporal modifica la forma en que las ondas generadas por cada elemento se suman en el espacio.

1.6.1. Concepto de *beamforming*

El término *beamforming* describe las técnicas que permiten conformar o dirigir un haz utilizando varios elementos [5, 6]. En el caso más sencillo, si todos los transductores emiten a la vez, el haz se orienta según la geometría natural del array. En cambio, si entre elementos consecutivos se introduce un pequeño retardo, el frente de onda resultante cambia y el haz puede inclinarse hacia una dirección determinada.

De forma intuitiva, el *beamforming* consiste en controlar los tiempos de activación de los distintos elementos para conseguir que sus contribuciones se sumen de forma constructiva en una dirección concreta. Por ello, la capacidad de imponer retardos bien definidos entre canales es la base física de esta técnica.

1.6.2. Separación entre elementos

En un array, la distancia entre elementos consecutivos es una magnitud importante. Si los elementos están demasiado separados, pueden aparecer direcciones secundarias no deseadas, conocidas como lóbulos de rejilla. Para reducir este efecto, es habitual utilizar como criterio de diseño una separación del orden de media longitud de onda:

$$p \approx \frac{\lambda}{2}$$

donde p es la separación entre elementos y λ es la longitud de onda. A su vez, la longitud de onda se define como:

$$\lambda = \frac{c}{f}$$

donde c es la velocidad de propagación del sonido en el medio y f es la frecuencia de operación del transductor.

Este criterio no debe entenderse como una ley absoluta, sino como una regla de diseño habitual para favorecer que el lóbulo principal tenga un comportamiento adecuado y para reducir la aparición de lóbulos de rejilla dentro del rango angular de interés.

1.6.3. Retardo entre elementos adyacentes

Si se desea orientar el haz un ángulo θ , el retardo entre dos elementos consecutivos puede relacionarse con la geometría del array mediante la diferencia de camino recorrida por el frente de onda. Para dos elementos separados una distancia p , esta diferencia de camino puede expresarse como:

$$\Delta l = p \sin \theta$$

Por tanto, el retardo temporal entre elementos adyacentes viene dado por:

$$\Delta t = \frac{\Delta l}{c} = \frac{p \sin \theta}{c}$$

donde Δt es el retardo entre elementos adyacentes, p es la separación entre elementos, c es la velocidad del sonido en el medio y θ es el ángulo de direccionamiento.

La Figura 1.2 muestra la interpretación geométrica de este retardo. En ella se representa un frente de onda plano incidente sobre un array lineal, donde la diferencia de camino entre elementos consecutivos genera el retardo temporal necesario para orientar el haz.

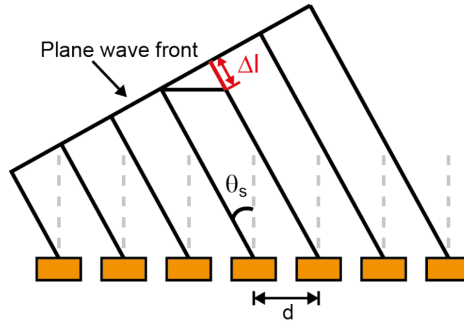


Figura 1.2: Relación geométrica entre la separación de los elementos del array, la diferencia de camino y el ángulo de direccionamiento del frente de onda. En la figura, d representa la separación entre elementos, equivalente a p en la notación utilizada en este trabajo. [6]

Despejando el ángulo, se obtiene:

$$\theta = \arcsin \left(\frac{\Delta t c}{p} \right)$$

Es decir, una vez conocido el retardo temporal entre elementos consecutivos, puede estimarse el ángulo hacia el que se orienta el haz. El signo del retardo dependerá de la convención utilizada y del sentido en el que se quiera dirigir la secuencia de activación.

Si se adopta la aproximación $p = \lambda/2$, con $\lambda = c/f$, entonces:

$$p = \frac{c}{2f}$$

Sustituyendo en la expresión anterior se llega a:

$$\theta = \arcsin(2f\Delta t)$$

Esta forma resulta especialmente útil porque relaciona directamente la frecuencia de operación del transductor con el retardo temporal impuesto por la electrónica.

1.6.4. Relación con una cadena digital de retardos

En una implementación digital, los retardos no suelen variar de forma continua, sino en pasos discretos definidos por el reloj del sistema. Si se dispone de una cadena de biestables o *flip-flops* activados por flanco, cada paso temporal adicional introduce un retardo igual a un periodo de reloj:

$$T_{clk} = \frac{1}{f_{clk}}$$

donde f_{clk} es la frecuencia del reloj interno u operativo.

En el caso particular estudiado en este trabajo, la cadena dispone de siete pasos temporales disponibles. Por tanto, el retardo máximo programable puede expresarse como:

$$\Delta t_{\text{máx}} = 7T_{clk} = 7\left(\frac{1}{f_{clk}}\right)$$

Si este valor máximo se sustituye en la expresión del ángulo, se obtiene:

$$\theta_{\text{máx}} = \arcsin(2f\Delta t_{\text{máx}})$$

o, de forma equivalente:

$$\theta_{\text{máx}} = \arcsin\left(2f\frac{7}{f_{clk}}\right)$$

Esta ecuación resume la relación entre la parte física y la parte electrónica del problema. Por un lado, aparece la frecuencia del transductor f , que depende del dispositivo ultrasónico. Por otro, aparece la frecuencia de reloj f_{clk} , que depende de la arquitectura digital. Finalmente, el número de pasos temporales útiles fija el retardo máximo disponible. Todo ello determina el rango de direccionamiento que el sistema puede alcanzar, siempre que el argumento del arcoseno se mantenga dentro del intervalo válido:

$$|2f\Delta t| \leq 1$$

1.7. Contexto del proyecto

Este trabajo se enmarca en el desarrollo de sistemas ultrasónicos integrados realizado en el ECAS Group (*Electronic Circuits and Systems Research Group*), utilizando tecnología CMOS HV de SilTerra de $0.13\ \mu\text{m}$ [7, 8, 3, 6, 9, 10].

El interés del proyecto no se limita a emitir una señal ultrasónica, sino que se centra en controlar de forma programable la relación temporal entre varios canales. Esta temporización es la que permite implementar estrategias básicas de direccionamiento del haz, por lo que la arquitectura digital de retardos constituye una pieza central del sistema.

La introducción desarrollada en este capítulo permite ver que dicha arquitectura no surge de manera arbitraria. Responde a una necesidad física concreta: si se desea orientar un haz ultrasónico mediante varios elementos, es imprescindible imponer retardos temporales entre ellos. El diseño electrónico que se estudiará en los capítulos posteriores debe entenderse, por tanto, como la traducción circuital de ese requisito físico.

Con esta base, en los siguientes capítulos se abordará la arquitectura concreta desarrollada en el trabajo, su funcionamiento y su implementación desde el punto de vista del diseño electrónico.

1.8. Organización del proyecto

La memoria se estructura de forma progresiva para ir desde la base física del problema hasta la validación funcional del sistema completo. En el Capítulo 1 se introducen los conceptos necesarios para entender el trabajo: ultrasonidos, PMUTs, arrays y relación entre retardos temporales y *beamforming*. En el Capítulo 2 se recogen los requisitos funcionales principales del sistema, como el número de canales, la frecuencia de reloj, el número de pasos temporales disponibles y el rango de barrido asociado.

El Capítulo 3 describe la arquitectura propuesta a nivel de sistema, diferenciando la fase de programación, la fase de transmisión y la estructura interna de los canales. El Capítulo 4 presenta la implementación a nivel de transistor de las celdas digitales utilizadas, junto con sus simulaciones DC y temporales. El Capítulo 5 muestra la implementación física mediante *layout*, la tecnología empleada y el proceso de verificación física realizado sobre los bloques principales. El Capítulo 6 valida funcionalmente el comportamiento del sistema completo y, finalmente, el Capítulo 7 recoge las conclusiones generales del trabajo.

Capítulo 2

Especificaciones y planteamiento del sistema

Una vez presentada la base física del *beamforming*, este capítulo define las especificaciones principales del sistema desarrollado [5]. El objetivo no es describir todavía la arquitectura interna bloque a bloque, sino fijar los parámetros funcionales que cumple el circuito: número de canales, frecuencia de reloj requerida, retardo programable, modo de configuración, señal de transmisión, tensión de salida y rango angular alcanzable.

El sistema se plantea como una arquitectura digital de generación de retardos programables para un array de PMUTs de ocho canales. La función principal del circuito es controlar la temporización relativa entre las salidas, de forma que sea posible modificar la dirección efectiva del haz emitido por los transductores.

En este apartado no se considera la frecuencia de reloj como un valor estrictamente fijo, sino como una variable de diseño. Dicha frecuencia se calcula a partir de la frecuencia de operación del PMUT disponible y del ángulo máximo de barrido que se desee obtener. Una vez determinada la frecuencia de reloj, queda fijado el periodo de reloj y, por tanto, el paso temporal del sistema:

$$T_{clk} = \frac{1}{f_{clk}}$$

El retardo entre canales se define de forma discreta mediante un número entero de pasos temporales, representado por la variable n . Este parámetro no puede tomar valores fraccionarios, sino únicamente valores enteros entre 0 y 7:

$$n \in \{0, 1, 2, 3, 4, 5, 6, 7\}$$

El valor $n = 0$ corresponde al caso sin retardo y $n = 7$ corresponde al mayor retardo programable disponible. De esta forma, el retardo teórico asociado a una configuración determinada queda relacionado con el periodo de reloj mediante nT_{clk} .

La frecuencia final de operación de los PMUTs no queda fijada de forma única en este punto del

trabajo. Por este motivo, se estudian varias frecuencias representativas: 1 MHz, 2 MHz, 5 MHz, 10 MHz y 20 MHz. Como caso de referencia se utiliza un PMUT de 5 MHz, ya que permite obtener un barrido cercano a $\pm 45^\circ$ con una frecuencia de reloj próxima a 100 MHz.

2.1. Especificaciones generales del sistema

La Tabla 2.1 resume las especificaciones generales consideradas para el sistema.

Tabla 2.1: Especificaciones generales del sistema.

Parámetro	Valor especificado
Número de canales	8
Frecuencia de reloj	Calculada según f_{PMUT} y $\theta_{\text{máx}}$
Paso temporal mínimo	$T_{\text{clk}} = 1/f_{\text{clk}}$
Retardo seleccionable entre canales	0 a 7 periodos de reloj
Retardo mínimo	0
Retardo máximo	$7T_{\text{clk}}$
Posiciones teóricas de direccionamiento	Hasta 15
Tensión final de salida	32 V
Separación temporal entre pulsos de transmisión	1 ms
Palabra de configuración	8 bits
Condición de carga de programación	Captura de 8 flancos útiles

El sistema dispone de ocho salidas de alta tensión, una por canal. Estas salidas están destinadas a excitar los PMUTs. La temporización relativa entre canales se define mediante una palabra de configuración de 8 bits, cargada en serie a través de la señal `DATA_IN`.

En este trabajo se ha comprobado el funcionamiento temporal del circuito para las frecuencias de reloj necesarias en los casos de diseño considerados. En particular, los casos estudiados cubren PMUTs entre 1 MHz y 20 MHz, con ángulos máximos de referencia de 45° y 60° . Por tanto, el análisis de especificaciones se centra en calcular la frecuencia de reloj requerida para cada caso y verificar que se mantiene dentro del rango de operación validado.

2.2. Frecuencia de reloj y pasos temporales disponibles

A partir de la relación teórica introducida en el capítulo anterior, y considerando una separación aproximada entre elementos de $d \approx \lambda/2$, el ángulo de barrido puede expresarse como:

$$\theta_n = \arcsin(2f_{\text{PMUT}}\Delta t_n)$$

El retardo digital aplicado entre canales se expresa como:

$$\Delta t_n = nT_{clk}$$

donde n representa el número de pasos temporales utilizados. Como:

$$T_{clk} = \frac{1}{f_{clk}}$$

se obtiene:

$$\Delta t_n = \frac{n}{f_{clk}}$$

Sustituyendo en la expresión del ángulo:

$$\theta_n = \arcsin\left(\frac{2f_{PMUT}n}{f_{clk}}\right)$$

Esta ecuación permite relacionar directamente la frecuencia del PMUT, la frecuencia de reloj y el número de pasos temporales seleccionados.

Para dimensionar el reloj a partir de un ángulo máximo deseado, se despeja f_{clk} :

$$\sin(\theta_{\text{máx}}) = \frac{2f_{PMUT}n}{f_{clk}}$$

$$f_{clk} = \frac{2nf_{PMUT}}{\sin(\theta_{\text{máx}})}$$

El criterio principal de especificación consiste en utilizar el mayor valor disponible de n , es decir:

$$n = 7$$

De esta forma, la frecuencia de reloj requerida queda:

$$f_{clk} = \frac{14f_{PMUT}}{\sin(\theta_{\text{máx}})}$$

Esta es la expresión principal utilizada para calcular la frecuencia de reloj necesaria en función de la frecuencia del PMUT y del ángulo máximo de barrido deseado.

El ángulo $\theta_{\text{máx}}$ queda libre en función de la aplicación. Teóricamente puede tomar valores entre:

$$0^\circ < \theta_{\text{máx}} \leq 90^\circ$$

Sin embargo, en la práctica se consideran más razonables valores máximos de barrido comprendidos aproximadamente entre 45° y 60° . Por este motivo, en este apartado se utilizan ambos valores como referencias de diseño.

2.2.1. Pasos temporales disponibles y número de posiciones

El parámetro n indica el número de pasos temporales utilizados para generar la separación temporal entre canales. En el sistema propuesto puede tomar los valores:

$$n = 0, 1, 2, 3, 4, 5, 6, 7$$

El valor $n = 0$ corresponde al caso sin retardo, mientras que $n = 7$ corresponde al mayor retardo programable disponible.

Para un valor máximo $n_{\text{máx}}$, el número total de posiciones discretas de steering que puede obtenerse considerando ambos sentidos de propagación es:

$$N_{\text{pos}} = 2n_{\text{máx}} + 1$$

El término $+1$ corresponde a la posición central, es decir, al caso $n = 0$, donde no se introduce retardo entre canales.

Por tanto, si se utiliza el valor máximo del sistema:

$$n_{\text{máx}} = 7$$

se obtiene:

$$N_{\text{pos}} = 2 \cdot 7 + 1 = 15$$

Estas 15 posiciones corresponden a siete ángulos en un sentido, una dirección central y siete ángulos en el sentido contrario. Por este motivo, desde el punto de vista de número de posiciones disponibles, lo más conveniente es dimensionar el sistema para poder utilizar $n = 7$.

Si la aplicación lo requiere, el valor máximo de n puede reducirse. Por ejemplo:

$$n_{\text{máx}} = 6 \Rightarrow N_{\text{pos}} = 13$$

$$n_{\text{máx}} = 5 \Rightarrow N_{\text{pos}} = 11$$

Sin embargo, al reducir n también se reduce el número de posiciones discretas disponibles para

formar el barrido. Por ello, el criterio preferente en las especificaciones es mantener $n = 7$, ya que permite obtener el máximo número de posiciones de steering.

2.2.2. Frecuencia de reloj requerida para distintos PMUTs

La Tabla 2.2 muestra la frecuencia de reloj requerida para distintas frecuencias de PMUT, considerando $n = 7$ y dos ángulos máximos de referencia: 45° y 60° .

Tabla 2.2: Frecuencia de reloj requerida utilizando $n = 7$.

Frecuencia PMUT	f_{clk} para $\theta_{\text{máx}} = 45^\circ$	f_{clk} para $\theta_{\text{máx}} = 60^\circ$
1 MHz	19.80 MHz	16.17 MHz
2 MHz	39.60 MHz	32.33 MHz
5 MHz	98.99 MHz	80.83 MHz
10 MHz	197.99 MHz	161.66 MHz
20 MHz	395.98 MHz	323.32 MHz

Puede observarse que el caso de 5 MHz encaja especialmente bien con una frecuencia de reloj cercana a 100 MHz si se desea obtener un barrido máximo próximo a 45° . Para PMUTs de mayor frecuencia, la frecuencia de reloj requerida aumenta proporcionalmente. La Tabla 2.2 se utiliza como referencia de dimensionamiento, mientras que la validación temporal del circuito debe comprobarse posteriormente con las simulaciones correspondientes.

2.3. Barrido angular para un PMUT de 5 MHz

Como caso de referencia se considera un PMUT de:

$$f_{\text{PMUT}} = 5 \text{ MHz}$$

Si se desea un ángulo máximo aproximado de 45° utilizando $n = 7$, la frecuencia de reloj requerida viene dada por:

$$f_{clk} = \frac{14 f_{\text{PMUT}}}{\sin(\theta_{\text{máx}})}$$

$$f_{clk} = \frac{14 \cdot 5 \text{ MHz}}{\sin(45^\circ)}$$

$$f_{clk} \approx 98.99 \text{ MHz}$$

Por tanto, una frecuencia de reloj práctica de 100 MHz permite aproximar este caso de diseño.
Con:

$$f_{clk} = 100 \text{ MHz}$$

se obtiene:

$$\theta_n = \arcsin\left(\frac{2 \cdot 5 \text{ MHz} \cdot n}{100 \text{ MHz}}\right)$$

$$\theta_n = \arcsin(0.1n)$$

La Tabla 2.3 muestra el barrido angular completo que se obtiene combinando los valores de n con el bit de dirección D7.

Tabla 2.3: Barrido angular completo para $f_{\text{PMUT}} = 5 \text{ MHz}$ y $f_{clk} = 100 \text{ MHz}$.

Sentido de propagación	D7	n	Retardo Δt_n	Ángulo θ_n
Inverso	1	7	$7T_{clk} = 70 \text{ ns}$	-44.43°
Inverso	1	6	$6T_{clk} = 60 \text{ ns}$	-36.87°
Inverso	1	5	$5T_{clk} = 50 \text{ ns}$	-30.00°
Inverso	1	4	$4T_{clk} = 40 \text{ ns}$	-23.58°
Inverso	1	3	$3T_{clk} = 30 \text{ ns}$	-17.46°
Inverso	1	2	$2T_{clk} = 20 \text{ ns}$	-11.54°
Inverso	1	1	$1T_{clk} = 10 \text{ ns}$	-5.74°
Centro	X	0	$0T_{clk} = 0 \text{ ns}$	0.00°
Directo	0	1	$1T_{clk} = 10 \text{ ns}$	$+5.74^\circ$
Directo	0	2	$2T_{clk} = 20 \text{ ns}$	$+11.54^\circ$
Directo	0	3	$3T_{clk} = 30 \text{ ns}$	$+17.46^\circ$
Directo	0	4	$4T_{clk} = 40 \text{ ns}$	$+23.58^\circ$
Directo	0	5	$5T_{clk} = 50 \text{ ns}$	$+30.00^\circ$
Directo	0	6	$6T_{clk} = 60 \text{ ns}$	$+36.87^\circ$
Directo	0	7	$7T_{clk} = 70 \text{ ns}$	$+44.43^\circ$

Por tanto, para un PMUT de 5 MHz, el sistema permite obtener 15 direcciones discretas de barrido: siete en un sentido, una dirección central y siete en el sentido contrario. El rango angular teórico queda aproximadamente entre:

$$-44.43^\circ \leq \theta \leq +44.43^\circ$$

Este resultado es especialmente interesante, ya que el sistema se aproxima al objetivo intuitivo de un barrido de -45° a $+45^\circ$ utilizando una frecuencia de reloj práctica cercana a 100 MHz y los siete pasos temporales disponibles.

2.4. Gráficas de barrido para distintas frecuencias de PMUT

Para estudiar el comportamiento del sistema con distintas frecuencias de PMUT, se han generado cinco gráficas. En cada una de ellas se fija la frecuencia del PMUT y se representa el ángulo de barrido en función de la frecuencia de reloj para los valores $n = 1, \dots, 7$.

El eje horizontal representa la frecuencia de reloj, entre 60 MHz y 100 MHz. El eje vertical representa el ángulo de barrido calculado mediante:

$$\theta_n(f_{clk}) = \arcsin\left(\frac{2f_{PMUT}n}{f_{clk}}\right)$$

Cada curva corresponde a un valor distinto de n . Cuando una curva no aparece en parte del rango, significa que para esa combinación de f_{PMUT} , f_{clk} y n , el argumento de la función arco seno supera la unidad y, por tanto, no existe un ángulo real válido.

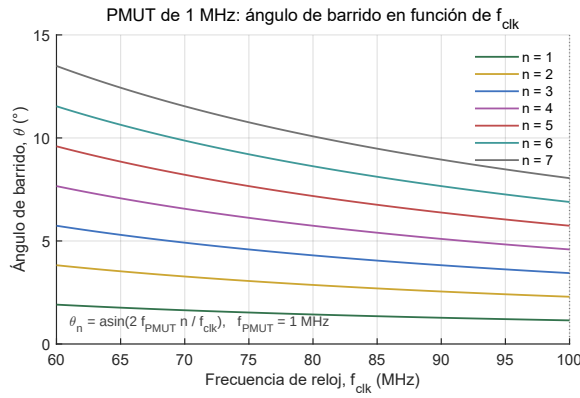


Figura 2.1: Ángulo de barrido en función de f_{clk} para un PMUT de 1 MHz.

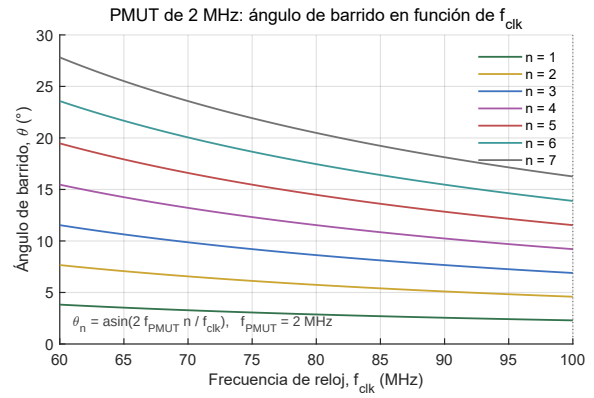


Figura 2.2: Ángulo de barrido en función de f_{clk} para un PMUT de 2 MHz.

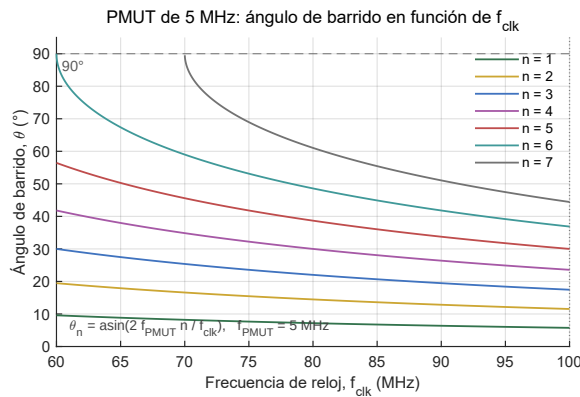


Figura 2.3: Ángulo de barrido en función de f_{clk} para un PMUT de 5 MHz.

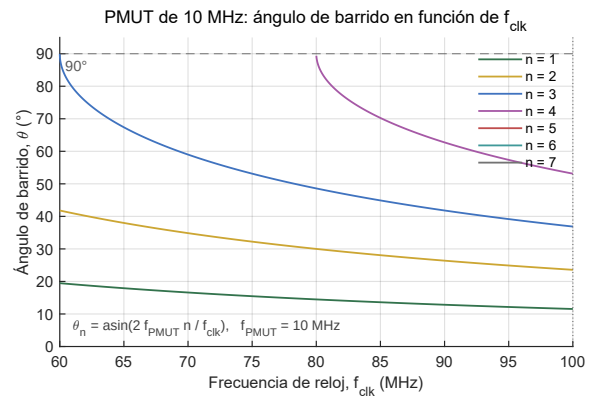


Figura 2.4: Ángulo de barrido en función de f_{clk} para un PMUT de 10 MHz.

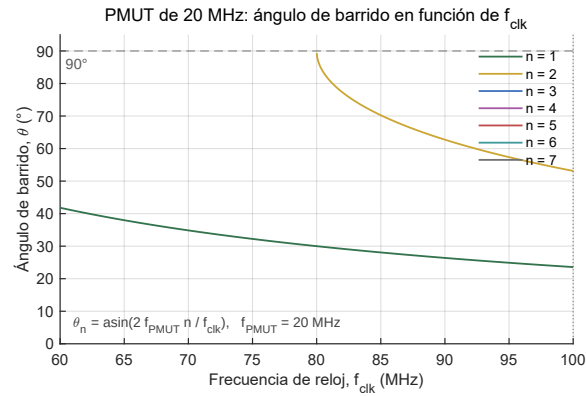


Figura 2.5: Ángulo de barrido en función de f_{clk} para un PMUT de 20 MHz.

Estas gráficas permiten visualizar de forma directa el efecto de la frecuencia del PMUT, la frecuencia de reloj y el número de pasos de retardo sobre el ángulo de barrido. En general, para maximizar el número de posiciones de steering interesa dimensionar el sistema para poder utilizar $n = 7$. No obstante, dicho valor debe cumplir siempre la condición de validez del arco seno.

Capítulo 3

Arquitectura general propuesta

Una vez definidas las especificaciones principales del sistema, en este capítulo se presenta la arquitectura general propuesta. El objetivo es describir la organización del circuito desde un punto de vista funcional, empezando por la visión global del sistema y bajando progresivamente hasta la estructura interna de un canal de transmisión.

El sistema se plantea como una arquitectura digital para excitar un array de PMUTs de ocho canales mediante retardos programables. Para ello, se utilizan tres señales externas:

CLK, CONF_EN, DATA_IN

La señal CLK proporciona la referencia temporal del sistema, CONF_EN separa la fase de programación de la etapa de transmisión, y DATA_IN tiene una doble función temporal. Durante la configuración transporta la palabra de control, mientras que durante la transmisión transporta los pulsos de transmisión, también conocidos como bursts, que se aplican al bloque multicanal.

En esta vista global también conviene fijar el papel de las señales de configuración. DATA_IN determina el número de pasos temporales que se aplican en la cadena de retardo y el sentido de propagación entre canales. Por tanto, la palabra de configuración no solo almacena un valor digital, sino que define directamente la temporización relativa de las salidas.

Desde el punto de vista funcional, la arquitectura puede dividirse en dos grandes bloques:

- un **bloque de configuración**, encargado de recibir la palabra de control y generar las señales internas necesarias para la operación;
- un **bloque de transmisión**, encargado de aplicar dicha configuración a los ocho canales y generar las salidas temporizadas hacia los PMUTs.

Esta organización permite separar claramente la carga de programación de la generación de señales de transmisión, evitando que la actividad de configuración se propague de forma no deseada hacia los canales.

3.1. Vista general de la arquitectura

A nivel funcional, el sistema completo puede representarse como la conexión entre un bloque de configuración y un bloque de transmisión. El primero recibe las señales externas del sistema y genera las señales internas de control, mientras que el segundo utiliza dichas señales para producir las salidas temporizadas hacia los transductores.

La Figura 3.1 muestra esta vista general de la arquitectura. En negro se representan las señales externas y la salida del sistema, mientras que en rojo se indican las señales internas que conectan la configuración con la transmisión.

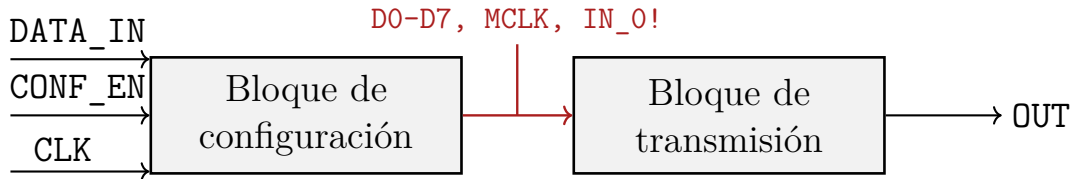


Figura 3.1: Vista general de la arquitectura propuesta a nivel de bloques principales.

Por tanto, primero se carga y se fija la configuración, y después dicha configuración gobierna la transmisión multicanal.

3.2. Codificación del retardo y selección de sentido

La arquitectura no solo se define por los bloques físicos que la forman, sino también por la forma en que se interpretan las señales de configuración. Por este motivo, la codificación del retardo y el bit de dirección se describen en este capítulo, ya que determinan directamente el funcionamiento interno del bloque generador de retardos y de la lógica de selección del canal.

Codificación del retardo mediante D0–D6

El retardo programado se define mediante el número de pasos temporales seleccionados. Este número se representa mediante la variable n :

$$n = 0, 1, 2, 3, 4, 5, 6, 7$$

El retardo asociado a cada valor de n viene dado por:

$$\Delta t_n = nT_{clk}$$

Como:

$$T_{clk} = \frac{1}{f_{clk}}$$

también puede escribirse como:

$$\Delta t_n = \frac{n}{f_{clk}}$$

Por tanto, n indica cuántos periodos de reloj se introducen como retardo entre canales. La Tabla 3.1 muestra la programación funcional del retardo mediante los bits D0–D6. La tabla se expresa una vez considerada la correspondencia interna de señales hacia la cadena de retardos.

En el diseño existe una correspondencia cruzada entre los bits almacenados en la lógica de configuración y las entradas aplicadas. En concreto, los bits asociados al retardo se aplican de forma invertida entre D0 y D6, mientras que D7 se mantiene como bit de dirección:

Antes de invertir	D0	D1	D2	D3	D4	D5	D6	D7
Después de invertir	D6	D5	D4	D3	D2	D1	D0	D7

Como se puede observar, se invierte la posición de los bits de retardo, exceptuando el bit D7, que conserva su posición porque se utiliza para seleccionar el sentido de propagación.

Tabla 3.1: Programación funcional del retardo mediante D0–D6.

D6	D5	D4	D3	D2	D1	D0	n	Retardo teórico
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	T_{clk}
0	0	0	0	0	1	1	2	$2T_{clk}$
0	0	0	0	1	1	1	3	$3T_{clk}$
0	0	0	1	1	1	1	4	$4T_{clk}$
0	0	1	1	1	1	1	5	$5T_{clk}$
0	1	1	1	1	1	1	6	$6T_{clk}$
1	1	1	1	1	1	1	7	$7T_{clk}$

Esta codificación no se interpreta como un número binario convencional, sino como una selección acumulativa de etapas. A mayor valor de n , mayor número de periodos de reloj se introducen como retardo.

Selección del sentido de propagación mediante D7

El bit D7 no modifica el valor del retardo. Su función es seleccionar el sentido de propagación de la secuencia temporal entre canales.

Por tanto, el valor de n controla el módulo del ángulo de barrido, mientras que D7 controla el signo del ángulo al invertir el orden temporal de activación de los canales.

De forma conceptual:

Tabla 3.2: Función del bit D7.

Valor de D7	Función
0	Propagación en sentido directo
1	Propagación en sentido inverso mediante la ruta REVERSE

$$n \Rightarrow |\theta|$$

$$D7 \Rightarrow \text{signo de } \theta$$

3.3. Fase de programación y fase de transmisión

Antes de entrar en el detalle interno de cada bloque, veremos la secuencia temporal general del sistema. El funcionamiento global se divide en dos etapas: durante la fase de programación se cargan los 8 bits de control, mientras que durante la etapa de transmisión se aplican los bursts hacia los canales.

La Figura 3.2 representa de forma conceptual esta separación. En negro se muestran las señales externas aplicadas al circuito y en rojo las señales internas generadas por la lógica del sistema.

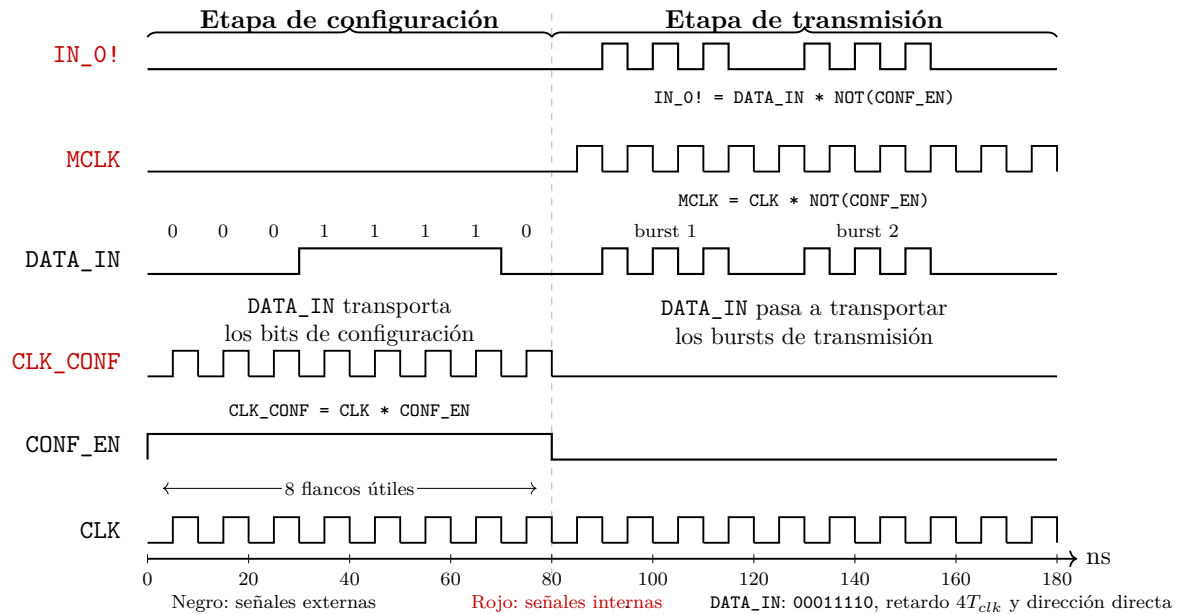


Figura 3.2: Separación temporal entre la fase de programación y la fase de transmisión. (Los pulsos a transmitir se representan de forma conceptual y no están dibujados a escala temporal real)

Durante la fase de programación, `CONF_EN` permanece activa. En ese intervalo, el reloj de configuración se genera como:

$$\text{CLK_CONF} = \text{CLK} \cdot \text{CONF_EN}$$

Por tanto, el registro serie-paralelo recibe los flancos necesarios para cargar la palabra de configuración. Al mismo tiempo, el reloj operativo permanece deshabilitado:

$$\text{MCLK} = \text{CLK} \cdot \overline{\text{CONF_EN}}$$

La entrada operativa hacia el primer canal también queda condicionada por `CONF_EN`. En concreto, `IN_0!` se genera bloqueando `DATA_IN` durante la fase de configuración y permitiendo su paso durante la fase de transmisión:

$$\text{IN_0!} = \text{DATA_IN} \cdot \overline{\text{CONF_EN}}$$

Cuando `CONF_EN` pasa a nivel bajo, la configuración queda fijada, `CLK_CONF` deja de proporcionar flancos útiles y `MCLK` se habilita como reloj operativo. A partir de ese momento, `DATA_IN` deja de interpretarse como palabra de configuración y pasa a transportar los pulsos de transmisión.

3.3.1. Bloque de configuración

El bloque de configuración agrupa funcionalmente el bloque `IN` que es el bloque habilitador de los relojes para la etapa de configuración y luego transmisión y la lógica serie-paralelo `CONF_LOGIC`. Su función es recibir las señales externas, separar la fase de programación de la fase de transmisión y generar las señales internas que utilizará posteriormente el bloque de transmisión.

La Figura 3.3 muestra el esquema conceptual del bloque de configuración. Las señales externas de entrada se representan en negro, mientras que las señales internas enviadas hacia la transmisión se representan en rojo.

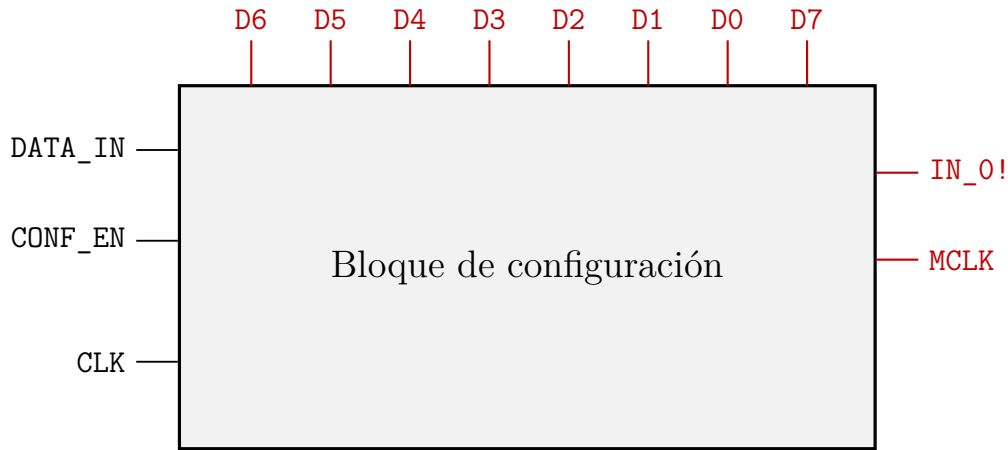
El bloque recibe como entradas externas:

$$\text{DATA_IN}, \quad \text{CONF_EN}, \quad \text{CLK}$$

y genera como señales internas principales:

$$\text{IN_0!}, \quad \text{MCLK}, \quad \text{D0-D7}$$

La señal `IN_0!` corresponde a la entrada operativa hacia el bloque multicanal, mientras que



Negro: señales externas

Rojo: señales internas hacia transmisión

Nota: CLK_CONF también se genera internamente, pero no se representa al no salir del bloque de configuración.

Figura 3.3: Esquema conceptual del bloque de configuración.

MCLK corresponde al reloj operativo de transmisión. Las señales D0–D7 forman la palabra de configuración almacenada. En la representación se muestran como D6, D5, D4, D3, D2, D1, D0 y D7, de acuerdo con la correspondencia interna utilizada por el diseño.

El bloque IN no actúa como una etapa de acondicionamiento analógico ni como una adaptación de nivel. Su función es puramente lógica: permite o bloquea el paso de determinadas señales según el estado de CONF_EN. Por otro lado, CONF_LOGIC implementa el registro serie-paralelo encargado de capturar la palabra de configuración.

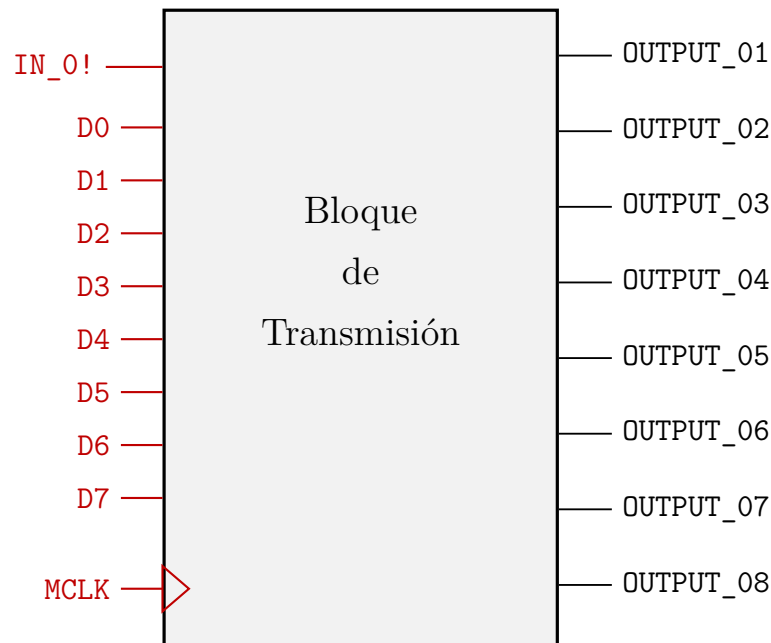
3.3.2. Bloque de transmisión

El bloque de transmisión recibe las señales generadas por la configuración y las utiliza para producir las ocho salidas finales hacia los PMUTs. Sus entradas principales son:

$$IN_0!, \quad MCLK, \quad D0-D7$$

La Figura 3.4 muestra el esquema conceptual del bloque de transmisión. Las señales procedentes de la configuración se muestran en rojo, mientras que las salidas hacia los PMUTs se representan en negro.

Desde el punto de vista arquitectónico, este bloque constituye la parte operativa del sistema. A partir de la señal de entrada y de la configuración almacenada, genera ocho señales de salida temporizadas, cada una asociada a uno de los canales del array.



Rojo: Procedentes de configuración Negro: salidas hacia los PMUTs

Figura 3.4: Esquema conceptual del bloque de transmisión.

3.4. Distribución conceptual de los ocho canales

El bloque de transmisión contiene ocho canales, denominados desde CH01 hasta CH08. Todos ellos comparten las señales de configuración y el reloj operativo, pero cada canal ocupa una posición distinta dentro de la cadena de propagación.

La Figura 3.5 muestra una representación conceptual conjunta de los ocho canales. Esta representación permite visualizar de forma directa cómo se encadenan los canales y cómo se distribuyen las señales de control comunes.

Esta organización muestra que los canales no trabajan como bloques independientes, sino como una estructura conectada. Las señales de propagación entre canales permiten establecer el orden temporal de activación y, mediante la configuración cargada, definir el retardo y el sentido de barrido.

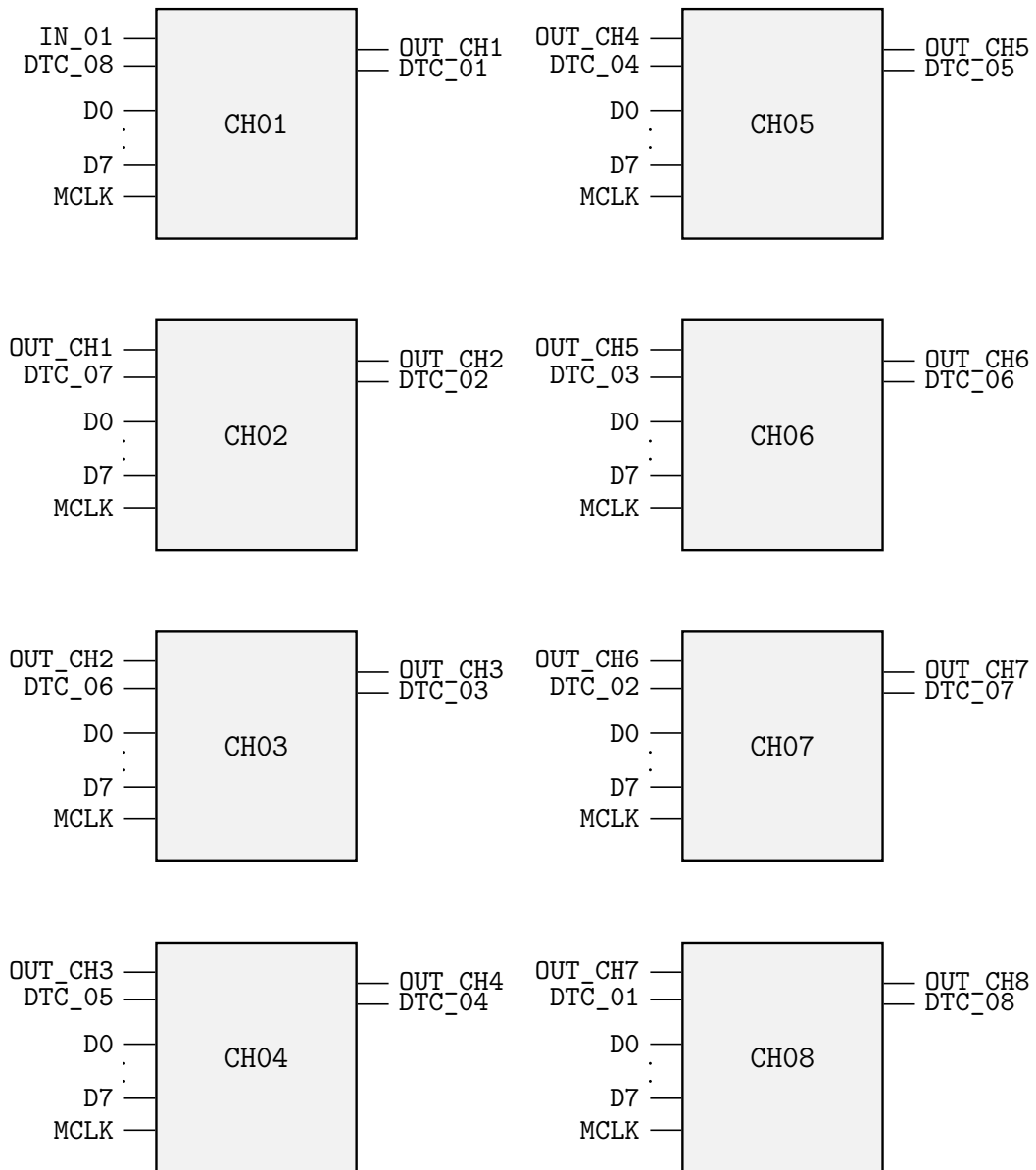


Figura 3.5: Distribución conceptual de los ocho canales dentro del bloque de transmisión.

3.5. Implementación circuital del canal y circuitería de selección de retardo

Cada canal constituye la unidad básica del bloque de transmisión. A nivel funcional, recibe una señal de entrada, el reloj operativo, la palabra de configuración y una señal de propagación procedente de la interconexión entre canales. A partir de estas señales genera una salida temporizada y elevada a alta tensión (32V).

La Figura 3.6 muestra el esquema funcional interno de un canal.

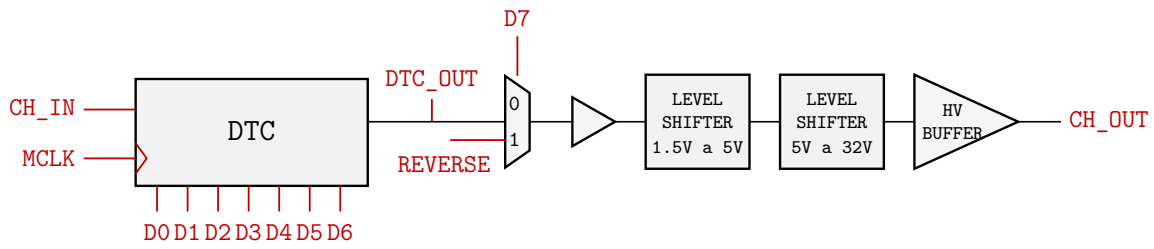


Figura 3.6: Esquema funcional interno de un canal de transmisión.

El primer bloque del canal es el DTC, encargado de generar el retardo programado a partir de la palabra de configuración D0–D6. La salida retardada, DTC_OUT, se aplica a un multiplexor junto con la señal REVERSE. Esta última corresponde a la ruta alternativa de propagación entre canales y, en cada canal, está asociada a la señal DTC_0X correspondiente.

El bit D7 controla el multiplexor y, por tanto, selecciona el sentido de propagación. A continuación, la señal seleccionada atraviesa un buffer, una primera adaptación de nivel de 1.5 V a 5 V, una segunda adaptación de 5 V a 32 V, y finalmente un buffer de alta tensión que entrega la salida del canal.

3.5.1. Cadena DTC y selección del retardo programado

El bloque DTC es el encargado de convertir la configuración digital de retardo en una separación temporal efectiva. Su estructura se basa en una cadena de siete flip-flops tipo D sincronizados mediante MCLK. Cada etapa de la cadena introduce un retardo de un periodo de reloj, de forma que el retardo total seleccionado puede expresarse como:

$$\Delta t_n = nT_{clk}$$

donde n es el número entero de pasos temporales habilitados, con:

$$n = 0, 1, 2, 3, 4, 5, 6, 7$$

La Figura 3.7 muestra una representación esquemática del funcionamiento interno del DTC. La señal de entrada CH_IN se propaga por la cadena de flip-flops y se generan distintas versiones

retardadas de la misma señal. Estas versiones se aplican a un multiplexor, que selecciona una de ellas como salida DTC_OUT .

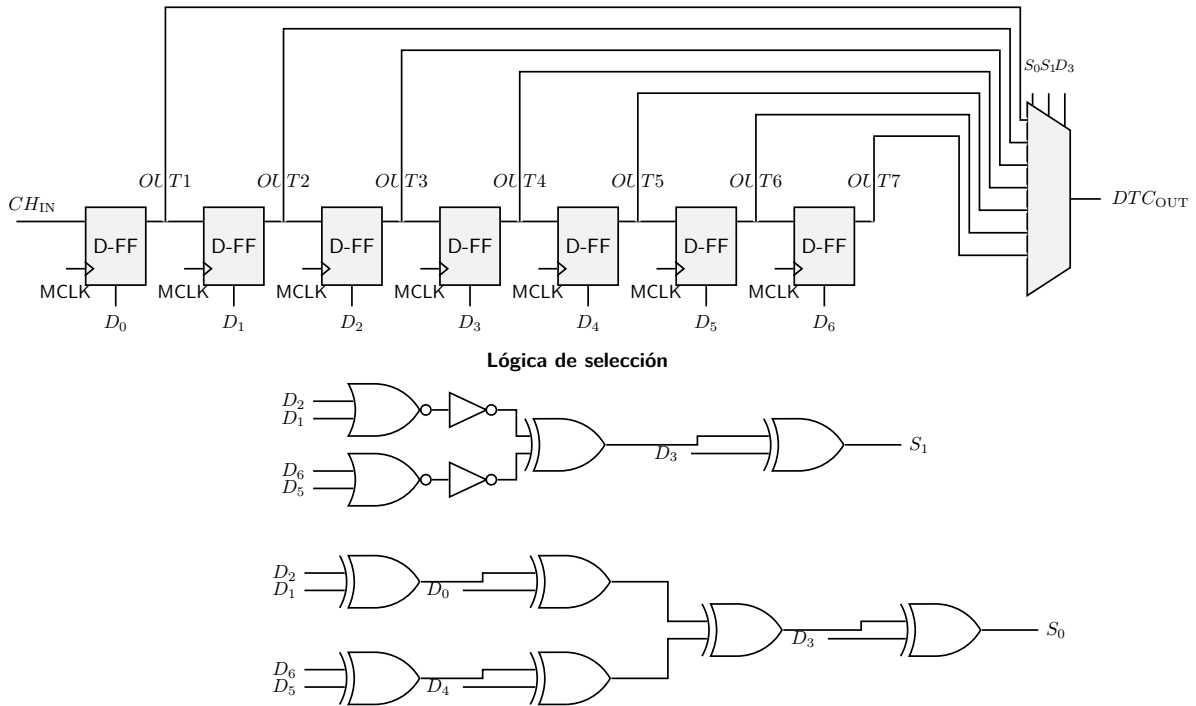


Figura 3.7: Estructura interna del DTC para la generación y selección del retardo programado.

El DTC genera retardo mediante una selección digital de diferentes versiones retardadas de la misma señal. La entrada CH_IN se aplica a una cadena de siete flip-flops. Cada flip-flop está sincronizado por $MCLK$, por lo que cada etapa introduce un retardo de un periodo de reloj.

Los bits $D0$ – $D6$ controlan los resets de los flip-flops. Estos resets son activos a nivel bajo, por lo que un valor lógico 0 mantiene la etapa correspondiente reseteada, mientras que un valor lógico 1 permite que dicha etapa funcione normalmente. Por tanto, la programación del retardo se interpreta como una habilitación acumulativa de etapas, no como un número binario convencional.

Además, debe tenerse en cuenta que los bits almacenados en el bloque serie-paralelo no llegan al DTC en el mismo orden físico en el que se cargan. En el diseño existe una inversión de los bits de retardo entre el bloque de configuración y el DTC:

$$D0 \rightarrow D6, \quad D1 \rightarrow D5, \quad D2 \rightarrow D4, \quad D3 \rightarrow D3, \quad D4 \rightarrow D2, \quad D5 \rightarrow D1, \quad D6 \rightarrow D0$$

De esta forma, una vez aplicada la inversión, el DTC recibe una codificación acumulativa coherente con el orden físico de la cadena de flip-flops. Por ejemplo, para obtener un retardo de $3T_{clk}$, las tres primeras etapas deben quedar habilitadas y el resto reseteadas:

$$D0 \ D1 \ D2 \ D3 \ D4 \ D5 \ D6 = 1110000$$

En este caso, D0, D1 y D2 valen 1, por lo que los tres primeros flip-flops funcionan normalmente. En cambio, D3–D6 valen 0, por lo que las etapas restantes permanecen reseteadas. La lógica de selección genera las señales necesarias para que el multiplexor tome la salida correspondiente al último flip-flop activo.

La Tabla 3.3 resume la codificación acumulativa vista desde la entrada real del DTC, una vez aplicada la inversión de bits.

Tabla 3.3: Codificación acumulativa del retardo vista desde el DTC.

n	D0	D1	D2	D3	D4	D5	D6	Retardo teórico
0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	$1T_{clk}$
2	1	1	0	0	0	0	0	$2T_{clk}$
3	1	1	1	0	0	0	0	$3T_{clk}$
4	1	1	1	1	0	0	0	$4T_{clk}$
5	1	1	1	1	1	0	0	$5T_{clk}$
6	1	1	1	1	1	1	0	$6T_{clk}$
7	1	1	1	1	1	1	1	$7T_{clk}$

Por tanto, el DTC puede entenderse como un selector de taps temporales. La cadena de flip-flops genera las posibles versiones retardadas de la señal, y el multiplexor selecciona la salida que corresponde al retardo programado. La salida final del bloque, DTC_OUT, se entrega posteriormente al multiplexor de dirección del canal, donde D7 decide si se utiliza la ruta directa o la ruta inversa de propagación entre canales.

Capítulo 4

Implementación a nivel de transistor

Las celdas descritas en este capítulo se han implementado y simulado en el entorno Cadence Virtuoso, utilizando simulaciones DC y transitorias para comprobar su comportamiento eléctrico [11].

4.1. Inversor CMOS

El inversor CMOS es la celda lógica más básica del sistema. Está formado por un transistor PMOS en la red *pull-up* y un transistor NMOS en la red *pull-down*. Ambas puertas están conectadas a la entrada *IN*, mientras que los drenadores de ambos transistores se unen en el nodo de salida *OUT*.

Cuando *IN* se encuentra a nivel bajo, el PMOS conduce y el NMOS permanece cortado, por lo que la salida se conecta a *DVDD*. En cambio, cuando *IN* está a nivel alto, el PMOS se corta y el NMOS conduce, conectando la salida a *DGND*. De esta forma, la celda implementa la función lógica:

$$\text{OUT} = \overline{\text{IN}}$$

Tabla 4.1: Dimensiones de los transistores del inversor CMOS.

Transistor	Finger width	Length	Number of fingers	Multiplier
PMOS	500 nm	150 nm	1	1
NMOS	200 nm	150 nm	1	1

En el inversor se ha utilizado una anchura de PMOS 2.5 veces mayor que la anchura de NMOS. Esta relación compensa, de forma aproximada, la menor movilidad de los portadores en el transistor PMOS y permite situar el punto de conmutación cerca de la condición $V_{in} = V_{out}$. Con este dimensionamiento, el cruce obtenido en la simulación DC queda próximo a la mitad de la alimentación, alrededor de 0.743 V.

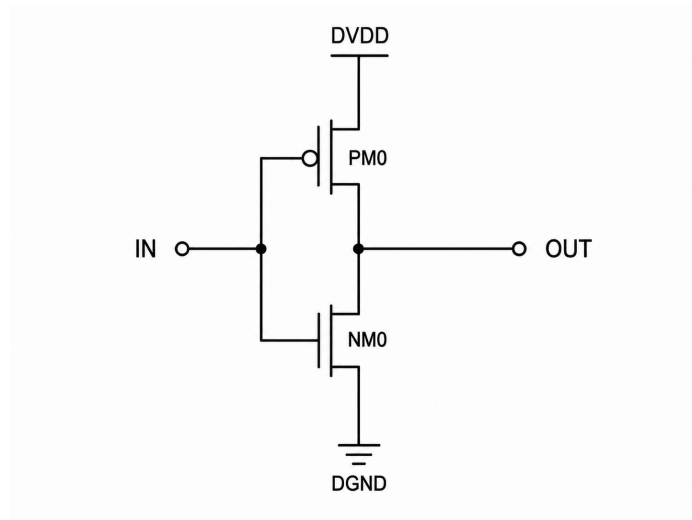


Figura 4.1: Implementación CMOS de un inversor.

La Figura 4.2 muestra la simulación DC del inversor. En ella se observa la transición de la salida desde nivel alto hasta nivel bajo a medida que aumenta la tensión de entrada. El punto de conmutación se define como la condición $V_{in} = V_{out}$, y en este caso se obtiene en 0.74316 V.

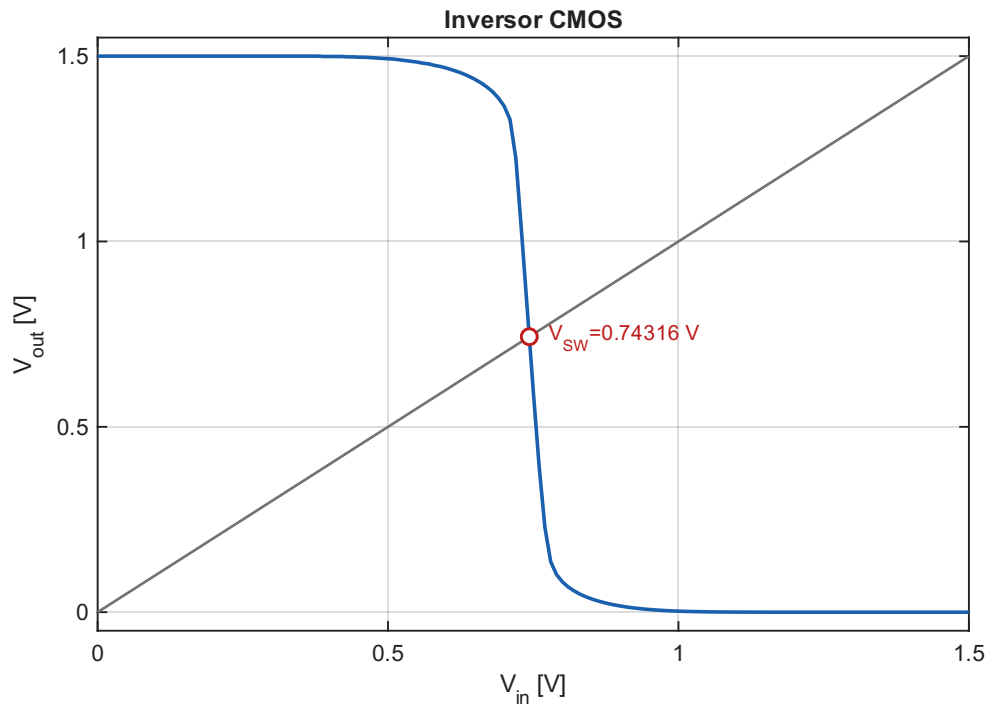


Figura 4.2: Simulación DC del inversor CMOS.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de subida y bajada.

Como se puede observar en la Figura 4.3, la salida conmuta en sentido contrario a la entrada.

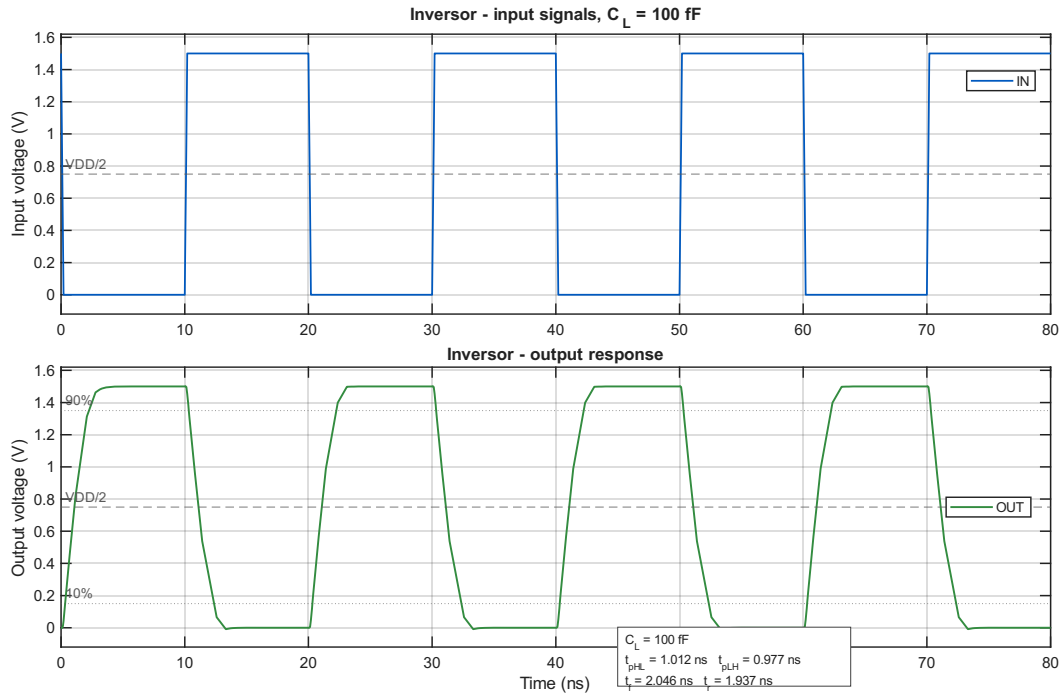


Figura 4.3: Respuesta temporal del inversor CMOS.

Por tanto, se cumple la tabla de verdad del inversor:

Tabla 4.2: Tabla de verdad del inversor CMOS.

IN	OUT
0	1
1	0

4.2. Buffer CMOS

El buffer CMOS se implementa mediante dos inversores conectados en cascada. El primer inversor genera una señal interna invertida, mientras que el segundo inversor vuelve a invertir dicha señal, recuperando la misma polaridad lógica que la entrada.

Por tanto, el buffer no modifica el valor lógico de la señal, sino que se utiliza para mejorar la capacidad de conducción, aislar etapas y reducir la carga efectiva vista por la celda anterior. Su función lógica puede expresarse como:

$$\text{OUT} = \text{IN}$$

En este caso, el segundo inversor del buffer se ha dimensionado con un número mayor de fingers respecto al primero. Esto permite aumentar la anchura efectiva de conducción de la etapa de salida, mejorando su capacidad para cargar y descargar nodos posteriores ya que circula una mayor intensidad debido a su inferior resistencia.

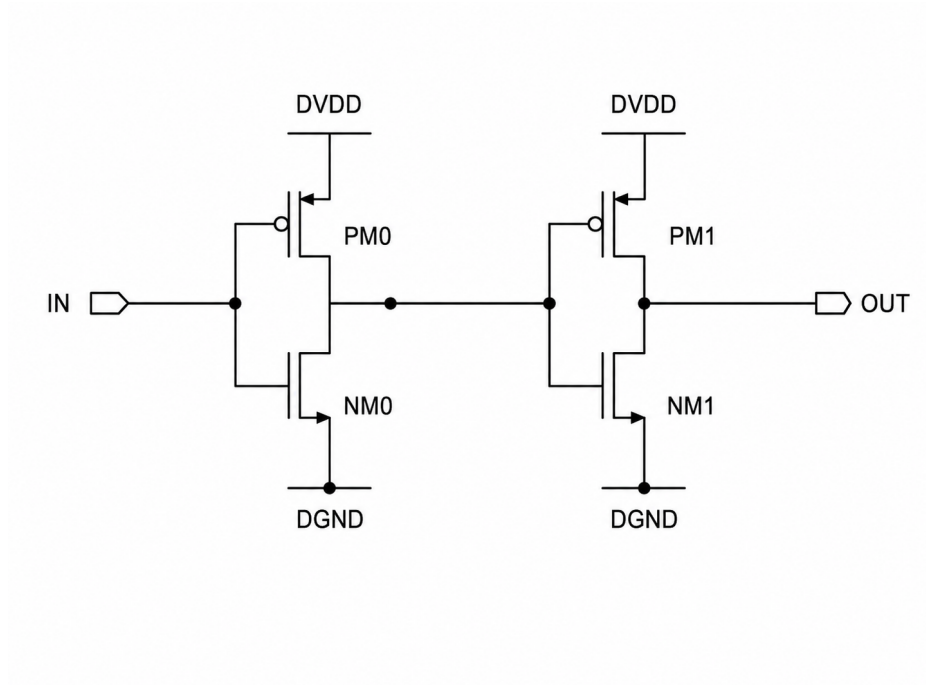


Figura 4.4: Implementación CMOS de un buffer mediante dos inversores en cascada.

Tabla 4.3: Dimensiones de los transistores del buffer CMOS.

Transistor	Finger width	Length	Number of fingers	Multiplier	Total width
PM0	600 nm	130 nm	1	1	600 nm
PM1	600 nm	130 nm	3	1	1.8 μ m
NM0	200 nm	130 nm	1	1	200 nm
NM1	200 nm	130 nm	3	1	600 nm

La simulación DC del buffer se muestra en la Figura 4.5. Al estar formado por dos inversores en cascada, la polaridad lógica final se conserva, aunque internamente existe una inversión intermedia. El punto de conmutación, definido por $V_{in} = V_{out}$, se obtiene en 0.75508 V.

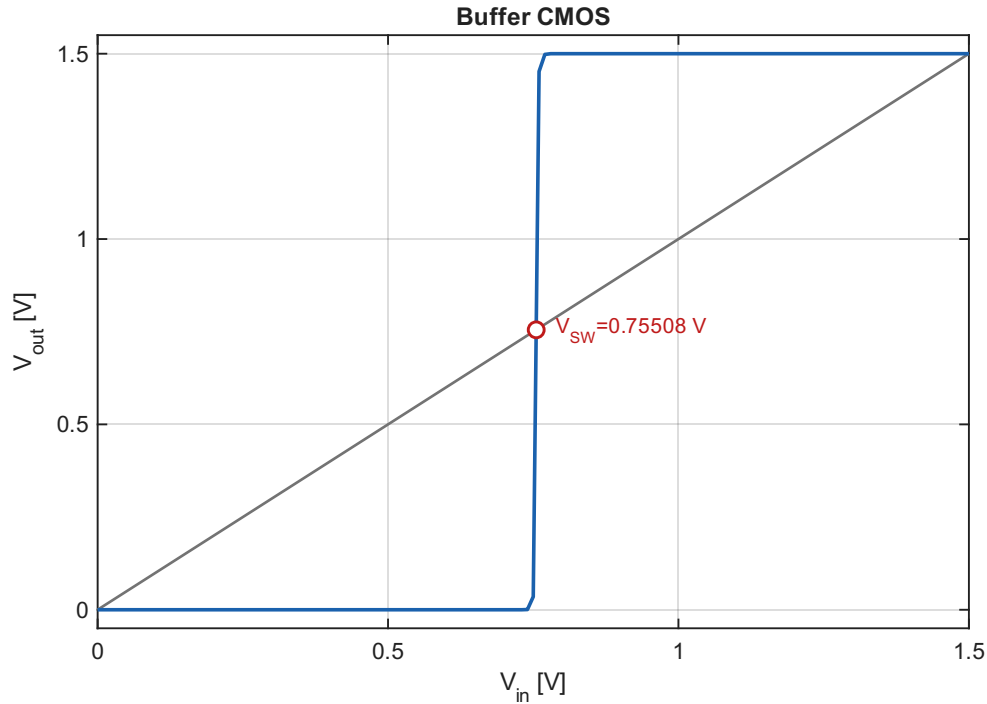


Figura 4.5: Simulación DC del buffer CMOS.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de subida y bajada.

Como se puede observar en la Figura 4.6, la salida mantiene la misma polaridad lógica que la entrada, aunque aparece un pequeño retardo debido a la propagación a través de los dos inversores internos. Por tanto, se cumple la tabla de verdad del buffer:

Tabla 4.4: Tabla de verdad del buffer CMOS.

IN	OUT
0	0
1	1

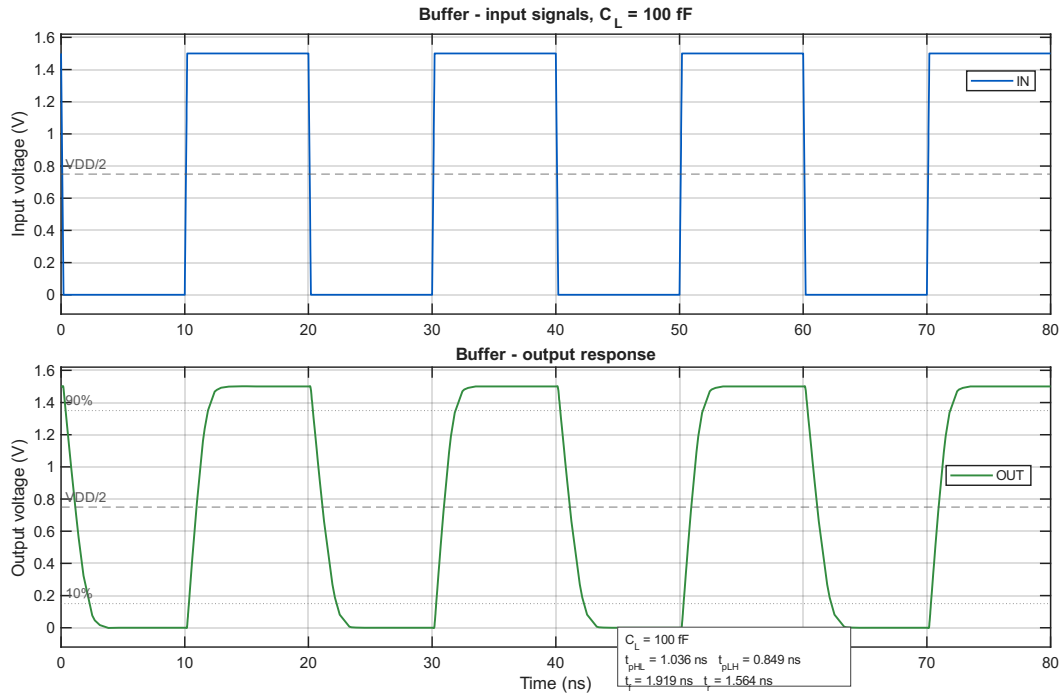


Figura 4.6: Respuesta temporal del buffer CMOS.

4.3. Puerta NAND CMOS de dos entradas

La puerta NAND de dos entradas se implementa mediante una red *pull-up* formada por dos transistores PMOS en paralelo y una red *pull-down* formada por dos transistores NMOS en serie.

En la parte superior, los dos PMOS comparten el nodo de alimentación DVDD en sus fuentes y el nodo de salida OUT en sus drenadores. La puerta del PMOS izquierdo está controlada por la señal A, mientras que la puerta del PMOS derecho está controlada por la señal B. Esta conexión en paralelo permite que la salida suba a nivel alto cuando al menos una de las entradas está a nivel bajo.

En la parte inferior, los dos NMOS están conectados en serie entre OUT y DGND. El NMOS superior está controlado por A, mientras que el NMOS inferior está controlado por B. Por tanto, la salida únicamente se descarga a masa cuando ambas entradas están a nivel alto. La función lógica implementada es:

$$OUT = \overline{A \cdot B}$$

La Figura 4.8 muestra la simulación DC de la puerta NAND de dos entradas. En esta simulación, el punto de conmutación definido por $V_{in} = V_{out}$ se obtiene en 0.82480 V.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de

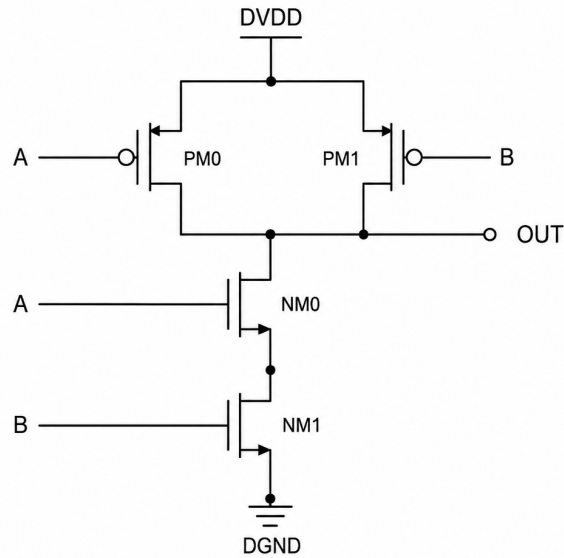


Figura 4.7: Implementación CMOS de una puerta NAND de dos entradas.

Tabla 4.5: Dimensiones de los transistores de la puerta NAND CMOS de dos entradas.

Transistor	Finger width	Length	Number of fingers	Multiplier
PM0	450 nm	150 nm	1	1
PM1	450 nm	150 nm	1	1
NM0	200 nm	150 nm	1	1
NM1	200 nm	150 nm	1	1

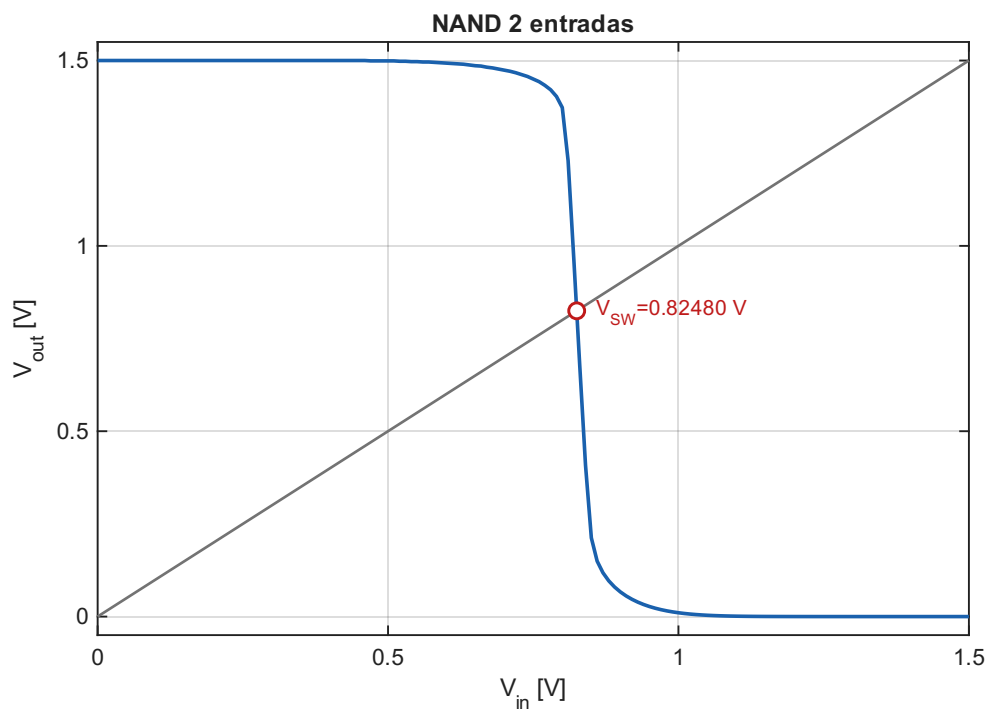


Figura 4.8: Simulación DC de la puerta NAND CMOS de dos entradas.

subida y bajada.

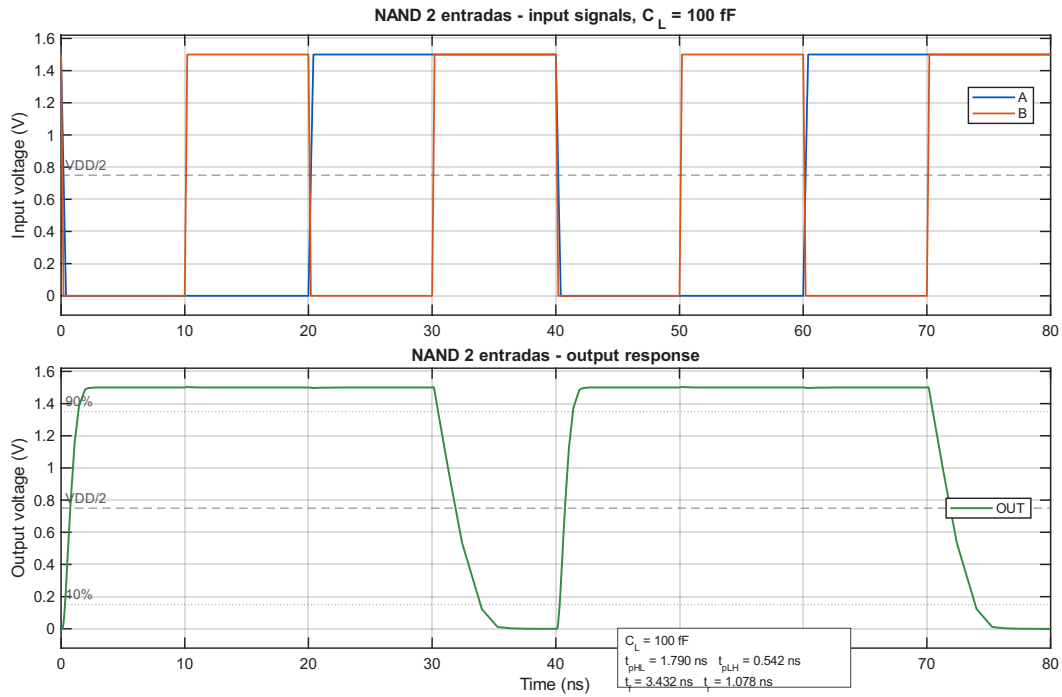


Figura 4.9: Respuesta temporal de la puerta NAND CMOS de dos entradas.

Como se puede observar en la Figura 4.9, la salida solo toma nivel lógico bajo cuando ambas entradas se encuentran a nivel lógico alto. Por tanto, se cumple la tabla de verdad de la puerta NAND de dos entradas:

Tabla 4.6: Tabla de verdad de la puerta NAND de dos entradas.

A	B	OUT
0	0	1
0	1	1
1	0	1
1	1	0

4.4. Puerta NOR CMOS de dos entradas

La puerta NOR de dos entradas se implementa mediante una red *pull-up* formada por dos transistores PMOS en serie y una red *pull-down* formada por dos transistores NMOS en paralelo.

En la parte superior, los dos PMOS se conectan en serie entre DVDD y el nodo de salida OUT. La puerta del transistor PM0 está controlada por la señal A, mientras que la puerta del transistor PM1 está controlada por la señal B. Debido a esta conexión en serie, la salida únicamente puede cargarse a nivel alto cuando ambas entradas están a nivel bajo.

En la parte inferior, los dos NMOS se conectan en paralelo entre OUT y DGND. El transistor NM0 está controlado por A, mientras que el transistor NM1 está controlado por B. Por tanto, si

cualquiera de las dos entradas se encuentra a nivel alto, la salida se descarga hacia masa. La función lógica implementada es:

$$OUT = \overline{A + B}$$

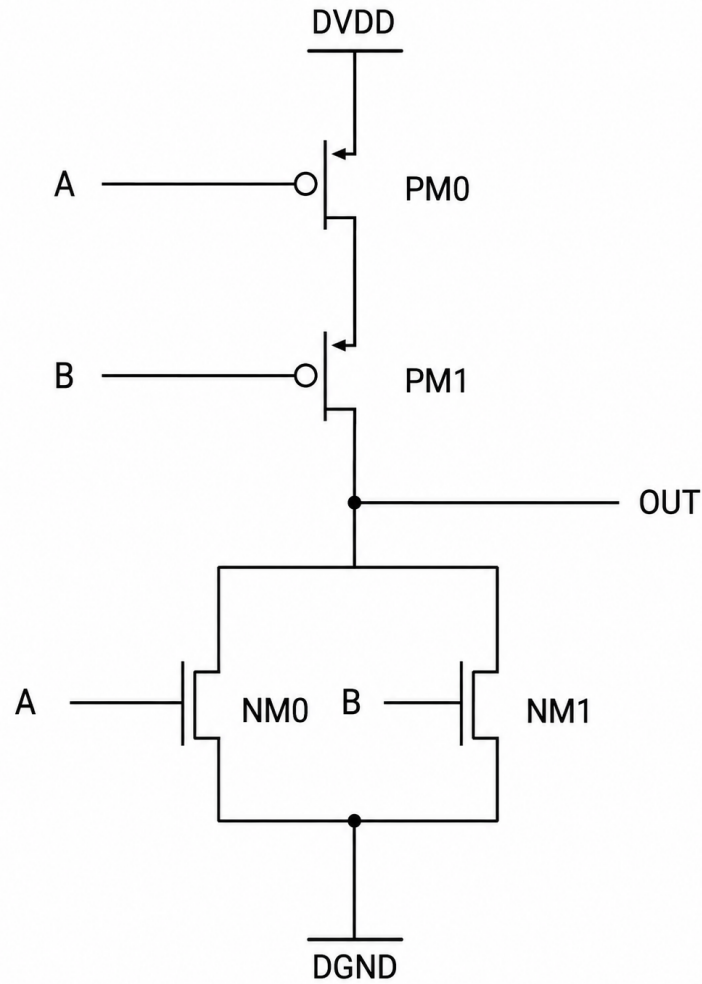


Figura 4.10: Implementación CMOS de una puerta NOR de dos entradas.

Tabla 4.7: Dimensiones de los transistores de la puerta NOR CMOS de dos entradas.

Transistor	Finger width	Length	Number of fingers	Multiplier
PM1	450 nm	150 nm	1	1
PM0	450 nm	150 nm	1	1
NM0	200 nm	150 nm	1	1
NM1	200 nm	150 nm	1	1

La simulación DC de la puerta NOR se utiliza para localizar el punto de conmutación de la celda. En este barrido se representa V_{out} en función de V_{in} , y el punto de conmutación se define como la condición $V_{in} = V_{out}$. Para esta puerta se obtiene 0.63844 V.

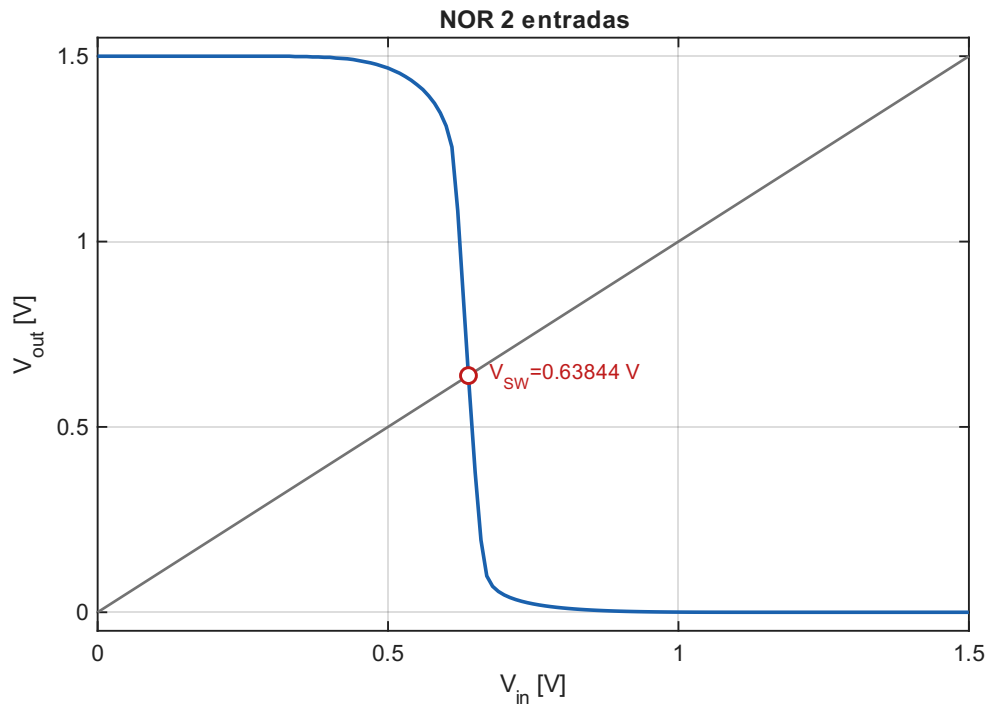


Figura 4.11: Simulación DC de la puerta NOR CMOS de dos entradas.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de subida y bajada.

Como se puede observar en la Figura 4.12, la salida únicamente toma nivel lógico alto cuando ambas entradas están a nivel bajo. Si cualquiera de las dos entradas pasa a nivel alto, la red *pull-down* conduce y la salida se descarga. Por tanto, se cumple la tabla de verdad de la puerta NOR de dos entradas:

Tabla 4.8: Tabla de verdad de la puerta NOR de dos entradas.

A	B	OUT
0	0	1
0	1	0
1	0	0
1	1	0

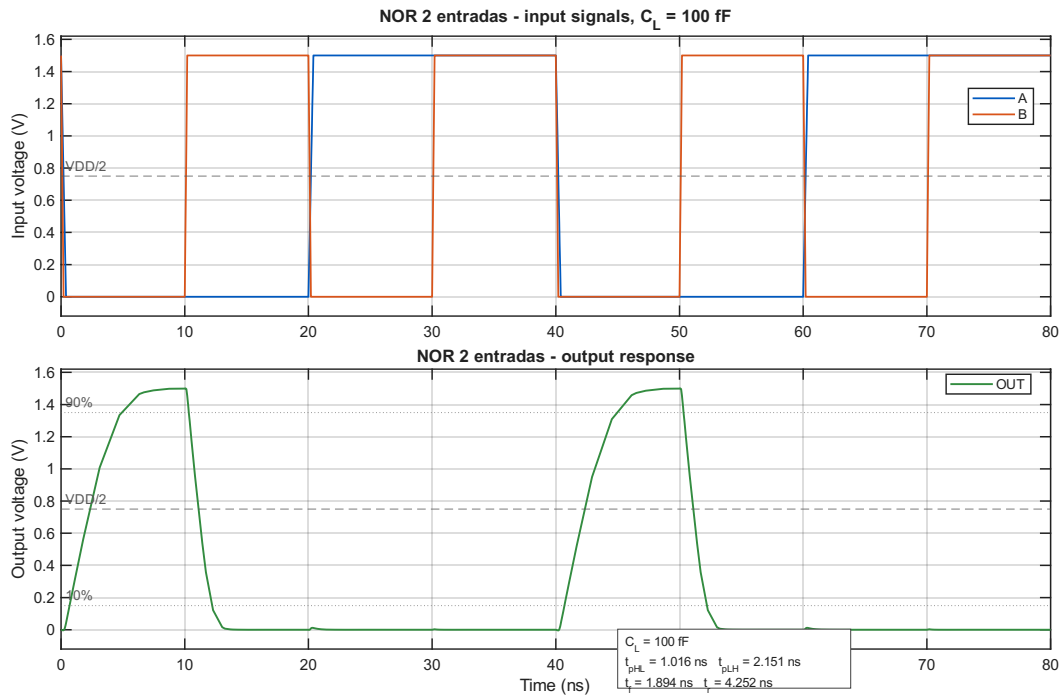


Figura 4.12: Respuesta temporal de la puerta NOR CMOS de dos entradas.

4.5. Puerta NAND CMOS de tres entradas

La puerta NAND de tres entradas sigue el mismo principio de funcionamiento que la NAND de dos entradas, pero ampliando el número de transistores de cada red. La red *pull-up* está formada por tres transistores PMOS en paralelo, mientras que la red *pull-down* está formada por tres transistores NMOS en serie.

Los transistores PM0 y NM0 están controlados por la señal A, PM1 y NM1 por la señal B, y PM2 y NM2 por la señal C. La salida únicamente se descarga a DGND cuando las tres entradas se encuentran a nivel alto. En cualquier otro caso, al menos uno de los PMOS conduce y la salida queda conectada a DVDD. La función lógica implementada es:

$$OUT = \overline{A \cdot B \cdot C}$$

Tabla 4.9: Dimensiones de los transistores de la puerta NAND CMOS de tres entradas.

Transistor	Finger width	Length	Number of fingers	Multiplier
PM0	450 nm	150 nm	1	1
PM1	450 nm	150 nm	1	1
PM2	450 nm	150 nm	1	1
NM0	200 nm	150 nm	1	1
NM1	200 nm	150 nm	1	1
NM2	200 nm	150 nm	1	1

La simulación DC de la NAND de tres entradas se muestra en la Figura 4.14. En este caso, el

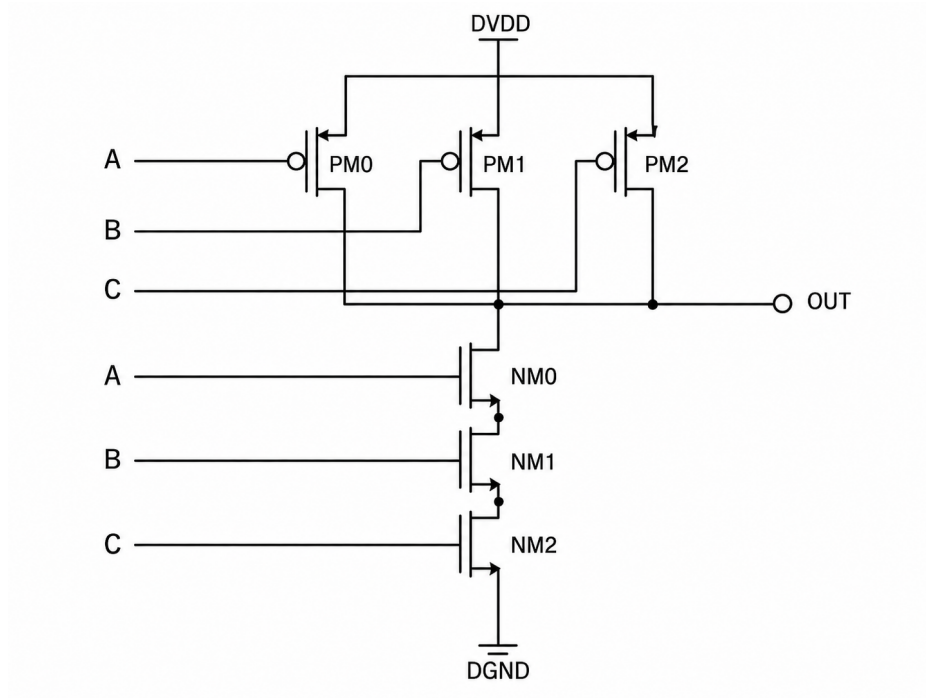


Figura 4.13: Implementación CMOS de una puerta NAND de tres entradas.

punto de conmutación definido por $V_{in} = V_{out}$ se obtiene en 0.87153 V.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de subida y bajada.

Como se puede observar en la Figura 4.15, la salida únicamente toma nivel lógico bajo cuando las tres entradas están simultáneamente a nivel alto. Por tanto, se cumple la tabla de verdad de la puerta NAND de tres entradas:

Tabla 4.10: Tabla de verdad de la puerta NAND de tres entradas.

A	B	C	OUT
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	0

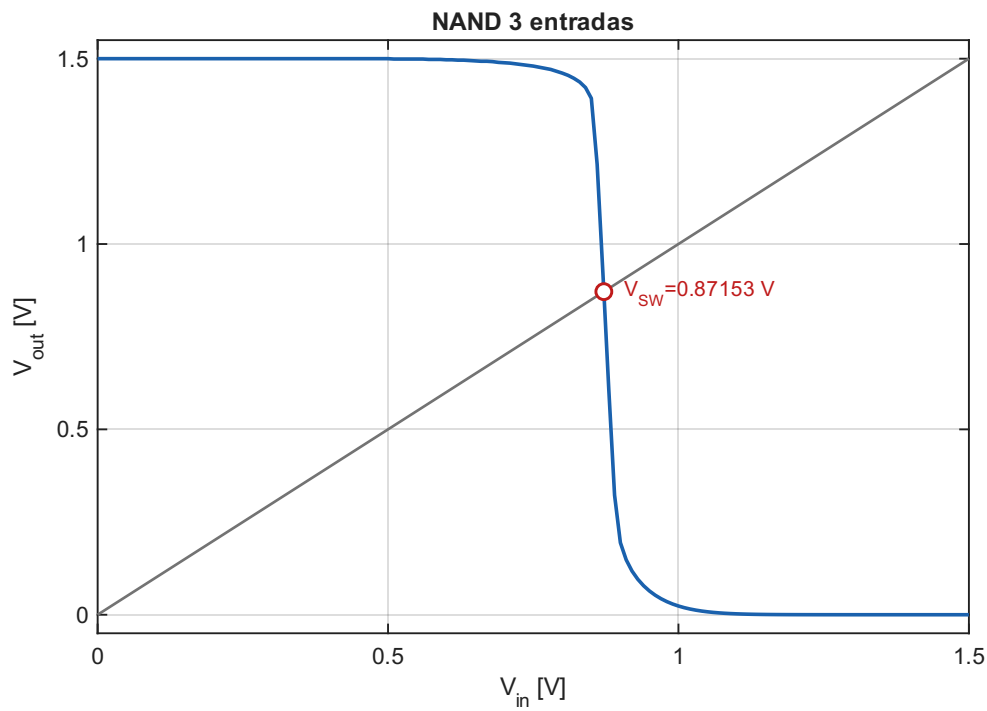


Figura 4.14: Simulación DC de la puerta NAND CMOS de tres entradas.

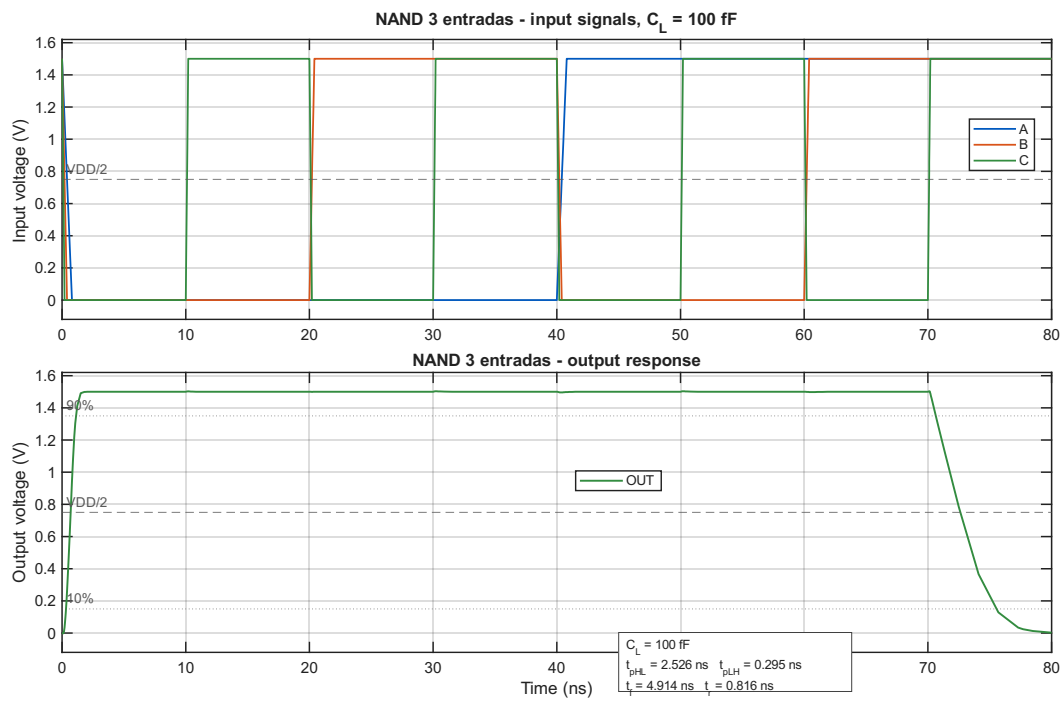


Figura 4.15: Respuesta temporal de la puerta NAND CMOS de tres entradas.

4.6. Puerta XOR CMOS

La puerta XOR implementa una función lógica de exclusividad entre dos entradas. La salida se activa cuando las entradas son diferentes y se desactiva cuando ambas entradas tienen el mismo valor lógico. Su función puede expresarse como:

$$\text{OUT} = A \oplus B$$

La implementación utilizada requiere las señales originales A y B, así como sus versiones complementarias A_N y B_N. Por este motivo, se incluyen dos inversores auxiliares encargados de generar dichas señales negadas.

La red principal está formada por ocho transistores. En la parte *pull-up*, los transistores PM0, PM1, PM2 y PM3 generan la conexión hacia DVDD en las combinaciones de entrada correspondientes. En la parte *pull-down*, los transistores NM0, NM1, NM2 y NM3 generan la descarga hacia DGND. Esta estructura permite implementar la función XOR completa utilizando lógica CMOS estática.

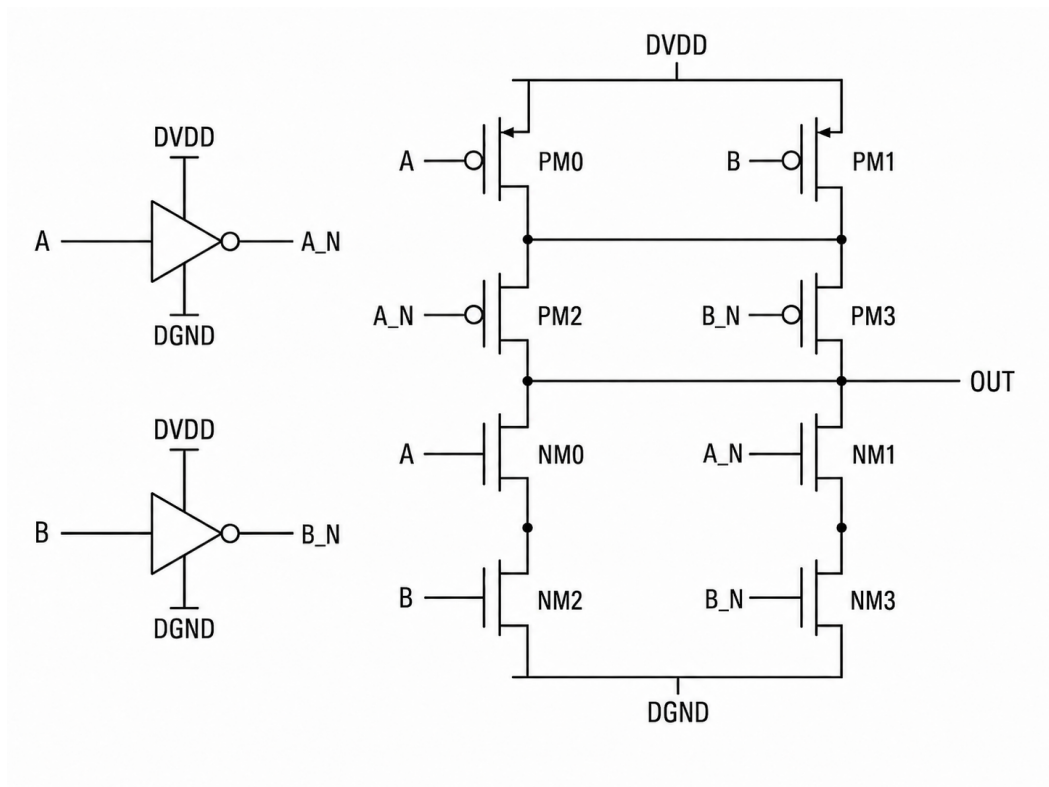
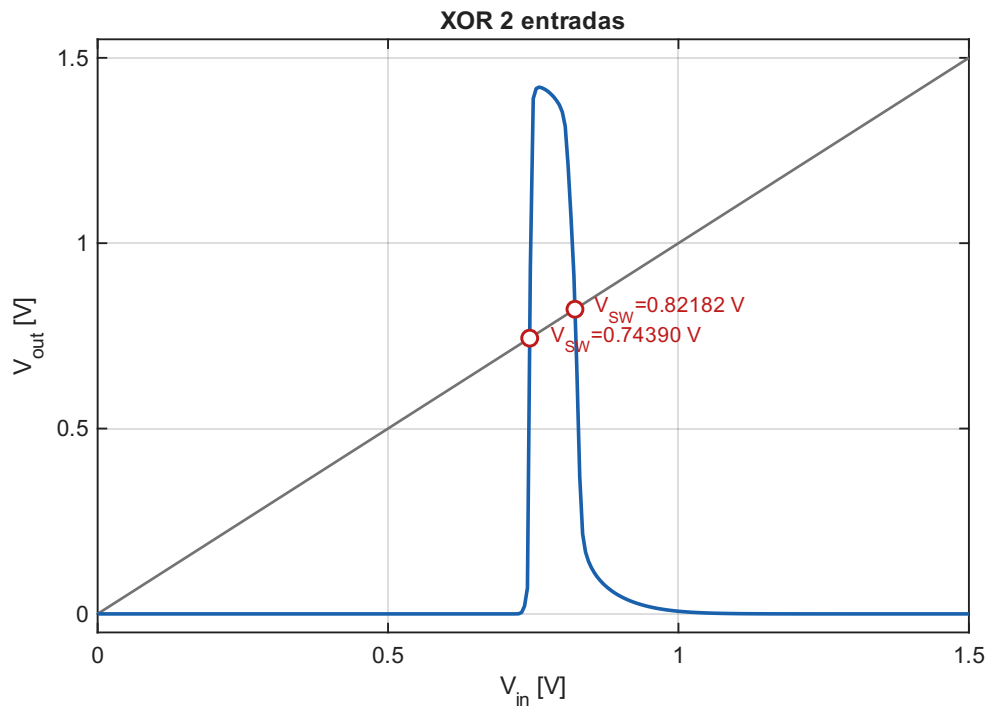


Figura 4.16: Implementación CMOS de una puerta XOR de dos entradas.

La Figura 4.17 muestra la simulación DC asociada a la puerta XOR. Para esta simulación se ha barrido la entrada correspondiente y se ha representado la salida resultante. Al tratarse de una celda no monótona en este barrido, aparecen dos puntos donde se cumple $V_{in} = V_{out}$: 0.74390 V y 0.82182 V.

Tabla 4.11: Dimensiones de los transistores de la puerta XOR CMOS.

Transistor	Finger width	Length	Number of fingers	Multiplier
PM0	450 nm	150 nm	1	1
PM1	450 nm	150 nm	1	1
PM2	450 nm	150 nm	1	1
PM3	450 nm	150 nm	1	1
NM0	200 nm	150 nm	1	1
NM1	200 nm	150 nm	1	1
NM2	200 nm	150 nm	1	1
NM3	200 nm	150 nm	1	1

**Figura 4.17:** Simulación DC de la puerta XOR CMOS.

Respuesta temporal con carga de salida. Para completar la caracterización, se representa la respuesta temporal de la celda con una carga capacitiva de salida de $C_L = 100$ fF. En esta gráfica se comprueba la función lógica y se extraen los retardos de propagación y los tiempos de subida y bajada.

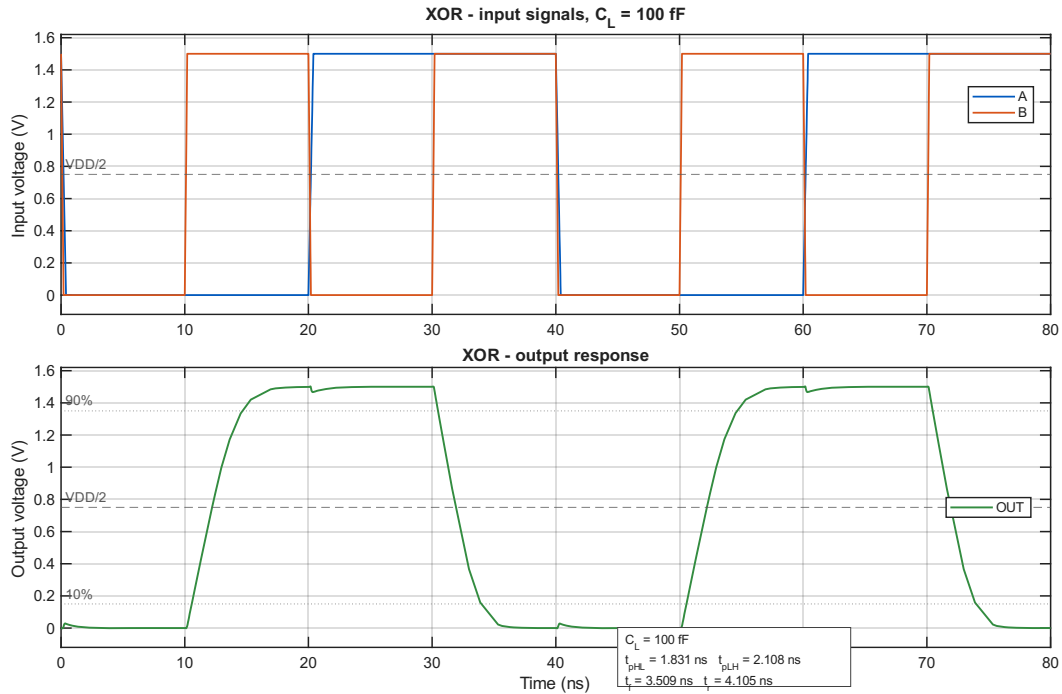


Figura 4.18: Respuesta temporal de la puerta XOR CMOS.

Como se puede observar en la Figura 4.18, la salida toma nivel lógico alto cuando las entradas son distintas y toma nivel lógico bajo cuando ambas entradas tienen el mismo valor lógico. Por tanto, se cumple la tabla de verdad de la puerta XOR:

Tabla 4.12: Tabla de verdad de la puerta XOR de dos entradas.

A	B	OUT
0	0	0
0	1	1
1	0	1
1	1	0

4.7. Flip-flop tipo D con reset asíncrono activo bajo

Una vez caracterizadas las puertas lógicas básicas, se analiza la celda secuencial utilizada en la cadena de retardo programable. El bloque empleado es un flip-flop tipo D activo por flanco, con reset asíncrono activo a nivel bajo. Esta celda constituye la unidad elemental de almacenamiento dentro del DTC, ya que cada paso temporal de la cadena introduce un retardo de un periodo de reloj.

La Figura 4.19 muestra la estructura lógica interna del flip-flop utilizado. La implementación se basa en puertas AND de tres entradas y realimentaciones internas que permiten conservar el estado lógico entre flancos de reloj. La celda recibe como entradas principales la señal de dato D, la señal de reloj CLK, la señal de reset **Reset** y la alimentación digital DVDD. Sus salidas son Q y Q', siendo esta última la señal complementaria de la salida principal.

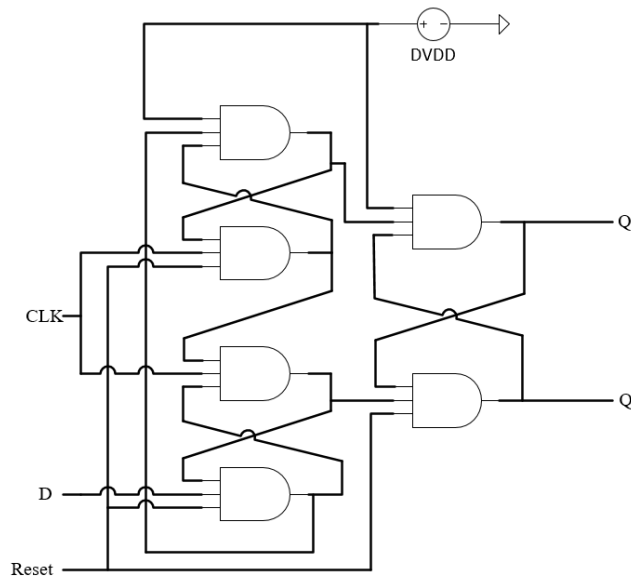


Figura 4.19: Implementación lógica del flip-flop tipo D utilizado en la cadena de retardo.

Desde el punto de vista funcional, el flip-flop captura el valor presente en la entrada D en el flanco activo de la señal de reloj. Una vez capturado, el valor queda almacenado en la salida Q hasta la llegada del siguiente flanco activo, siempre que la señal de reset permanezca inactiva. Por tanto, la función secuencial esperada puede expresarse como:

$$Q^+ = D$$

donde Q^+ representa el nuevo estado almacenado tras el flanco activo de reloj.

La señal **Reset** actúa de forma asíncrona y activa a nivel bajo. Esto significa que su efecto no depende del reloj: cuando **Reset** toma valor lógico bajo, la celda se fuerza directamente al estado de reset. En cambio, cuando **Reset** se mantiene a nivel alto, el flip-flop queda habilitado y opera normalmente como una celda tipo D.

Reset = 0 $\Rightarrow$ estado de reset

Reset = 1 $\Rightarrow$ funcionamiento normal

Esta característica es especialmente importante dentro del DTC. En la cadena de retardo programable, cada flip-flop dispone de una entrada de reset controlada por uno de los bits de configuración D0–D6. Como el reset es activo a nivel bajo, el bit de configuración no actúa como un número binario convencional, sino como una señal de habilitación de etapa:

$D_i = 1 \Rightarrow$ flip-flop habilitado

$D_i = 0 \Rightarrow$ flip-flop reseteado

Por tanto, la programación del retardo en el DTC se realiza habilitando de forma acumulativa los primeros pasos temporales de la cadena y manteniendo reseteadas las restantes. Si una etapa está habilitada, la señal puede propagarse a través de ella en el siguiente flanco activo de MCLK. Si está reseteada, su salida queda forzada al estado de reset y no contribuye al retardo seleccionado.

Cada flip-flop habilitado introduce un retardo de un periodo de reloj. De este modo, si se habilitan n etapas consecutivas, el retardo generado por la cadena puede aproximarse como:

$$\Delta t_n = nT_{clk}$$

4.7.1. Respuesta temporal del flip-flop

La respuesta temporal del flip-flop se ha representado considerando el reset inactivo, es decir, $\text{Reset}=1$. En la parte superior de la Figura 4.20 se muestran las señales CLK y D, mientras que en la parte inferior se representa la salida Q. La señal complementaria Q_N no se incluye porque no se utiliza en el camino funcional considerado.

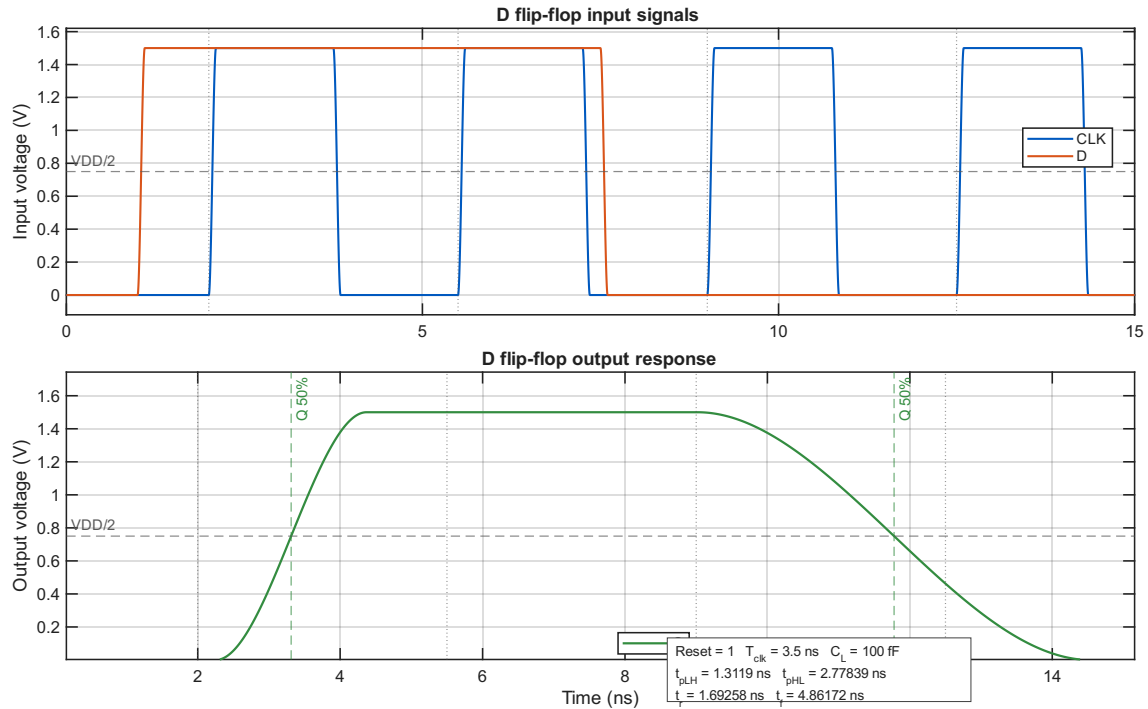


Figura 4.20: Respuesta temporal del flip-flop tipo D considerando $C_L = 100$ fF y $\text{Reset}=1$.

La salida Q actualiza su valor únicamente tras el flanco activo de reloj. Por tanto, los cambios en D no se trasladan directamente a la salida, sino que quedan sincronizados con CLK. Este comportamiento es precisamente el que permite utilizar una cadena de flip-flops como elemento básico para generar pasos temporales discretos dentro del DTC.

4.8. Level shifter de 1.5 V a 5 V

Después de la lógica digital de baja tensión, las señales deben adaptarse al dominio de 5 V antes de continuar hacia las etapas posteriores del canal. Para ello se utiliza un *level shifter* encargado de convertir una señal digital de entrada comprendida entre 0 V y 1.5 V en una señal de salida compatible con el dominio de 5 V.

El circuito utiliza transistores específicos de 5 V, correspondientes a los modelos `pmv_5v` y `nmv_5v`. Los PMOS se conectan a V_{DD_5V} , mientras que los NMOS permiten descargar los nodos internos hacia masa. La entrada IN procede del dominio digital de 1.5 V, y la salida OUT entrega una señal regenerada en el dominio de 5 V.

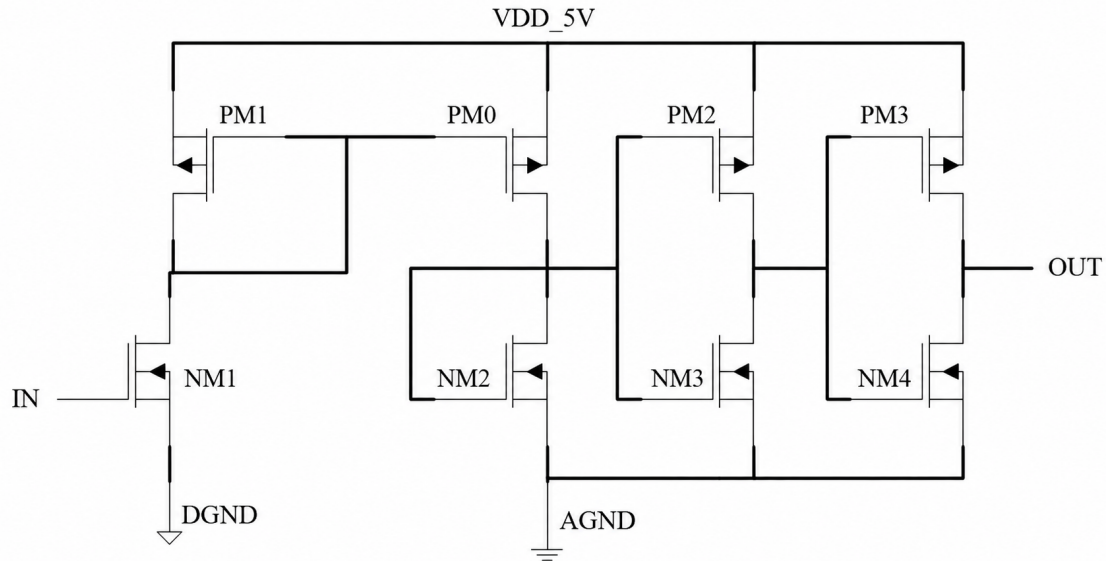


Figura 4.21: Implementación transistor-level del *level shifter* de 1.5 V a 5 V.

$$\text{IN} : 0 - 1.5 \text{ V} \longrightarrow \text{OUT} : 0 - 5 \text{ V}$$

El funcionamiento del bloque puede entenderse en dos etapas. En primer lugar, la estructura principal del *level shifter* desplaza el nivel lógico de entrada hacia el dominio de 5 V. Cuando la entrada se encuentra a nivel bajo, los nodos internos asociados permanecen próximos a masa y el circuito entrega un nivel lógico bajo. En cambio, cuando la entrada toma un valor lógico alto en el dominio de 1.5 V, el nodo interno previo a la etapa de salida se eleva progresivamente hasta alcanzar un nivel suficientemente alto dentro del dominio de 5 V, en torno a 4 V.

A continuación, este nodo intermedio se aplica a una etapa de salida formada por dos inversores conectados en cascada, que actúan como buffer. Aunque el nivel intermedio no alcance exactamente 5 V, un valor cercano a 4 V es interpretado por el primer inversor como un 1 lógico dentro del dominio de 5 V. La etapa de buffer regenera entonces la señal y entrega una salida rail-to-rail, obteniéndose finalmente una señal digital comprendida entre 0 V y 5 V.

De este modo, el bloque no actúa como un amplificador analógico, sino como una etapa digital de adaptación y regeneración de nivel. La señal de entrada de 1.5 V se convierte en una señal compatible con las etapas posteriores alimentadas a 5 V, manteniendo la información lógica original.

La Tabla 4.13 recoge las dimensiones de los transistores utilizados.

Este bloque constituye la primera etapa de adaptación de tensión entre la lógica digital de control y las etapas posteriores del canal. Su inclusión permite que las señales generadas por el DTC puedan propagarse correctamente hacia los bloques alimentados a mayor tensión.

Tabla 4.13: Dimensiones de los transistores del *level shifter* de 1.5 V a 5 V.

Transistor	Tipo	Modelo	Length	Number of fingers	Finger width	Total width
PM1	PMOS	pmv_5v	1 μm	1	10 μm	10 μm
NM1	NMOS	nmv_5v	1 μm	10	5 μm	50 μm
PM0	PMOS	pmv_5v	1 μm	5	10 μm	50 μm
NM2	NMOS	nmv_5v	1 μm	1	8 μm	8 μm
PM2	PMOS	pmv_5v	1 μm	1	7 μm	7 μm
NM3	NMOS	nmv_5v	1 μm	1	3 μm	3 μm
PM3	PMOS	pmv_5v	1 μm	3	7 μm	21 μm
NM4	NMOS	nmv_5v	1 μm	3	3 μm	9 μm

4.9. Resumen de parámetros temporales extraídos

A partir de las simulaciones temporales se han extraído los parámetros temporales principales de cada celda. El retardo t_{pHL} corresponde al tiempo de propagación de nivel alto a nivel bajo, mientras que t_{pLH} corresponde al tiempo de propagación de nivel bajo a nivel alto. Los tiempos t_f y t_r corresponden respectivamente al tiempo de bajada y al tiempo de subida de la salida.

Tabla 4.14: Parámetros temporales obtenidos mediante simulación temporal con $C_L = 100 \text{ fF}$.

Celda	t_{pHL} [ns]	t_{pLH} [ns]	t_f [ns]	t_r [ns]
Inversor	1.01151	0.97746	2.04600	1.93715
Buffer	1.03626	0.84889	1.91856	1.56373
NAND 2 entradas	1.78955	0.54159	3.43215	1.07760
NOR 2 entradas	1.01614	2.15062	1.89356	4.25213
NAND 3 entradas	2.52617	0.29450	4.91437	0.81592
XOR	1.83147	2.10785	3.50884	4.10477
Flip-flop tipo D	2.77839	1.31190	4.86172	1.69258

Para el flip-flop tipo D, la caracterización temporal se ha realizado considerando **Reset=1**, un reloj de ejemplo con $T_{clk} = 3.5 \text{ ns}$ y la misma carga capacitiva de salida $C_L = 100 \text{ fF}$.

Los resultados muestran que, al incluir una carga de 100 fF, los parámetros temporales se sitúan en el rango de nanosegundos. Las puertas con redes de transistores en serie presentan una resistencia equivalente mayor en determinadas transiciones, lo que incrementa especialmente los tiempos de bajada o subida según la topología. En el caso de la XOR, su estructura interna más compleja también se refleja en una respuesta temporal más lenta que la de puertas más simples.

4.10. Criterio de dimensionamiento

El dimensionamiento de las celdas digitales se ha realizado teniendo en cuenta la función de cada bloque dentro del sistema. En general, los transistores PMOS presentan una anchura

mayor que los NMOS para compensar parcialmente la menor movilidad de los portadores en el canal tipo p. Esto permite equilibrar en cierta medida la capacidad de carga y descarga de las celdas CMOS.

En el caso del inversor, se utiliza un PMOS de 500 nm de finger width y un NMOS de 200 nm, ambos con longitud de canal de 150 nm. Esta relación permite obtener una celda compacta, pero con una capacidad de conducción suficiente para regenerar niveles lógicos dentro de la arquitectura.

En el caso del buffer, la segunda etapa se dimensiona con tres fingers tanto para el PMOS como para el NMOS. Así, el transistor PM1 alcanza una anchura total de 1.8 μm , mientras que el transistor NM1 alcanza una anchura total de 600 nm. Esta segunda etapa actúa como etapa de salida del buffer, por lo que resulta razonable aumentar su tamaño respecto al primer inversor para mejorar la capacidad de conducción debido a la inferior resistencia por lo que mayor paso de corriente.

En las puertas NAND, los transistores PMOS se implementan con finger width de 450 nm, mientras que los NMOS utilizan finger width de 200 nm. En la red *pull-down*, los NMOS se encuentran en serie, lo que incrementa la resistencia equivalente durante la descarga de la salida. Este efecto es más acusado en la NAND de tres entradas, ya que la descarga se realiza a través de tres transistores NMOS conectados en serie.

En la puerta NOR, la red *pull-up* está formada por PMOS en serie, mientras que la red *pull-down* está formada por NMOS en paralelo. Por ello, las transiciones de subida pueden verse más penalizadas que en otras topologías, ya que la carga del nodo de salida debe realizarse a través de dos transistores PMOS en serie.

En la puerta XOR se mantiene el mismo criterio de dimensionamiento que en las puertas NAND para la red principal: PMOS de 450 nm y NMOS de 200 nm, con longitud de canal de 150 nm, un único finger y multiplicador igual a 1. Esto permite mantener una implementación homogénea dentro de las celdas lógicas combinacionales.

4.11. Resumen

Las celdas descritas en este apartado constituyen la base de la implementación digital del sistema. El inversor permite regenerar e invertir señales lógicas, el buffer permite reforzar la capacidad de conducción sin modificar la polaridad lógica, y las puertas NAND, NOR y XOR permiten construir funciones lógicas más complejas mediante lógica CMOS estática.

Las simulaciones DC permiten comprobar el punto aproximado de conmutación de las celdas, mientras que las simulaciones temporales permiten verificar su funcionalidad temporal y extraer parámetros como los retardos de propagación y los tiempos de subida y bajada.

Finalmente, se ha descrito la implementación lógica del flip-flop tipo D empleado en el DTC. Esta celda secuencial es esencial para la generación de retardos programables, ya que cada paso

temporal habilitado de la cadena introduce un retardo de un periodo de reloj. Su reset asíncrono activo a nivel bajo permite activar o desactivar cada etapa mediante los bits de configuración D0–D6, dando lugar a una programación acumulativa del retardo.

Estas celdas serán utilizadas posteriormente para construir bloques de mayor nivel, como el registro serie-paralelo de configuración, la lógica de selección del DTC, la cadena de retardo programable y el bloque completo de transmisión multicanal.

Capítulo 5

Implementación física mediante layout

Una vez definida la implementación esquemática y transistor-level de los bloques principales, el siguiente paso consiste en trasladar el circuito a una implementación física mediante *layout*. En esta fase ya no solo importa que la conectividad sea correcta, sino también cómo se colocan físicamente los transistores, cómo se distribuyen las alimentaciones, cómo se enrutan las señales y cuánto área acaba ocupando cada bloque.

El layout se ha realizado en Cadence Virtuoso, trabajando a partir de los esquemáticos previamente diseñados [11]. El objetivo ha sido obtener una implementación física coherente con el funcionamiento eléctrico del circuito, pero también razonablemente compacta. En un sistema multicanal como el propuesto, el área es un aspecto importante, ya que los bloques deben repetirse o integrarse dentro de una arquitectura de mayor tamaño.

Aunque en el apartado anterior se han explicado las puertas y celdas digitales básicas de forma separada, el proceso real de layout no se ha planteado siempre como una unión de bloques independientes colocados uno detrás de otro. En muchos casos, especialmente en bloques combinacionales como la lógica del DTC, se ha implementado directamente el módulo completo. Es decir, en lugar de diseñar por separado un inversor, una XOR, una AND, una NAND y después unir todas esas celdas, se ha colocado y optimizado el conjunto completo dentro del mismo layout.

Esta forma de trabajar permite reducir área, acortar interconexiones internas y aprovechar mejor el espacio disponible. Funcionalmente, el circuito puede describirse mediante puertas lógicas, pero físicamente resulta más eficiente compactar varios elementos dentro de un mismo bloque cuando estos forman parte de una misma función.

La excepción principal a este criterio han sido los flip-flops. Al tratarse de una celda secuencial repetitiva, utilizada tanto en el registro serie-paralelo como en la cadena de retardos del DTC, se ha diseñado una celda de flip-flop una vez y posteriormente se ha reutilizado mediante copia y colocación. En este caso sí tiene sentido mantener una celda repetida, ya que permite conservar

regularidad física, reducir errores y facilitar la conexión entre etapas.

En este apartado se documentan los layouts de las celdas y bloques principales desarrollados, así como la integración física de un canal completo y del sistema de ocho canales. De este modo, se puede observar la evolución desde las celdas digitales elementales hasta la implementación física global del bloque multicanal.

5.1. Proceso general de diseño físico

El diseño físico se ha realizado siguiendo el flujo habitual de implementación en Cadence Virtuoso. A partir de los esquemáticos validados, cada bloque se traslada a layout respetando las reglas de la tecnología, la conectividad eléctrica y la distribución de alimentaciones. Una vez dibujado el layout, el bloque se verifica mediante herramientas de Calibre de Mentor, pasando por las comprobaciones de DRC, LVS y extracción [12].

El DRC (*Design Rule Check*) permite comprobar que el layout cumple las reglas geométricas de la tecnología. El LVS (*Layout Versus Schematic*) verifica que el circuito dibujado físicamente corresponde al esquemático original. Finalmente, la extracción permite obtener una representación eléctrica del layout, incluyendo las conexiones físicas y, en fases posteriores, los efectos parásitos asociados.

Este flujo resulta esencial en una tecnología CMOS HV, ya que no basta con que el circuito sea correcto a nivel esquemático. También debe respetar las distancias, anchos, contactos, pozos y restricciones propias de los dispositivos de baja, media y alta tensión.

5.1.1. Tecnología CMOS HV SilTerra de 130 nm

La implementación se realiza sobre la PDK de SilTerra CMOS HV de 130 nm, correspondiente a la tecnología de SilTerra utilizada en el entorno de diseño [7, 8]. Esta tecnología permite integrar en un mismo circuito transistores de baja tensión, media tensión y alta tensión, lo que resulta especialmente útil para la arquitectura propuesta. La lógica digital principal trabaja en el dominio de baja tensión, mientras que las etapas finales deben alcanzar niveles de salida de hasta 32 V.

Dentro de la tecnología se utilizan transistores de baja tensión de 1.5 V, como los dispositivos NLV y PLV. También se dispone de dispositivos de media tensión, adecuados para dominios de 3.3 V, 5 V y 6 V, así como dispositivos de alta tensión para las etapas de salida de 32 V.

La tecnología dispone de seis capas de metal, que se utilizan para realizar el rutado interno de las celdas y las interconexiones entre bloques. Las capas metálicas permiten distribuir alimentaciones, señales de control, relojes y salidas, manteniendo una organización lo más compacta y clara posible.

5.1.2. Compactación de celdas, pozos y anillos de guarda

Para reducir el área ocupada por las celdas digitales, se ha intentado compartir regiones físicas siempre que ha sido posible. De forma general, los transistores NMOS se colocan en la parte inferior de la celda compartiendo pozo o región común, con su correspondiente polarización a masa. Los transistores PMOS se sitúan en la parte superior, también agrupados y polarizados adecuadamente hacia la alimentación.

Esta disposición permite compactar el layout y reducir la longitud de interconexiones internas. Además, facilita la distribución de los raíles de alimentación y masa, manteniendo una estructura regular entre celdas.

Al tratarse de una tecnología de alta tensión, se incorporan anillos de guarda alrededor de las regiones sensibles cuando resulta necesario. Estos anillos ayudan a aislar eléctricamente los dispositivos, mejorar la robustez frente a acoplos y mantener separadas las regiones de distinta tensión. Este criterio es especialmente importante en bloques que actúan como interfaz entre dominios de tensión diferentes, como los level shifters y las etapas de salida.

5.1.3. Criterios generales de layout

El layout de los bloques digitales se ha realizado manteniendo una organización clara de alimentación, masa, pozos y señales. En las celdas de baja tensión, la alimentación DVDD se distribuye habitualmente mediante un raíl superior, mientras que la referencia DGND se sitúa en la parte inferior. Esta distribución facilita la lectura del layout y permite conectar bloques de forma más ordenada.

Desde el punto de vista físico, las estructuras CMOS separan la zona de transistores PMOS de la zona de transistores NMOS. De forma general, los PMOS se colocan en la parte superior del layout, asociados a la red *pull-up*, mientras que los NMOS se colocan en la parte inferior, asociados a la red *pull-down*. Esta disposición mantiene una relación directa con la estructura transistor-level descrita previamente.

También se ha intentado reducir al máximo los cruces innecesarios entre señales. En bloques digitales pequeños esto resulta relativamente sencillo, pero en módulos con varias señales de control, como el DTC o su lógica de selección, el enrutado se vuelve más crítico. Por este motivo, en estos bloques se ha priorizado una implementación compacta del módulo completo, evitando una colocación excesivamente fragmentada en celdas independientes.

En los bloques repetitivos, como los flip-flops del registro de configuración y de la cadena DTC, sí se ha seguido una estrategia de reutilización directa. Una vez diseñado y comprobado el layout de la celda de flip-flop, se ha copiado en las distintas posiciones necesarias. Esta metodología permite mantener una estructura regular y facilita la distribución de señales comunes como el reloj y el reset.

Por tanto, la estrategia seguida puede entenderse como una metodología mixta: se mantiene

una jerarquía funcional clara, pero el layout físico se optimiza según las necesidades de cada bloque. Cuando una celda se repite muchas veces, se reutiliza. Cuando un bloque combinacional puede compactarse mejor como conjunto, se implementa directamente como bloque completo. Estos criterios de organización, compactación y cuidado en el enrutado son coherentes con las recomendaciones habituales de diseño físico en circuitos integrados [13].

Algunos bloques pertenecientes a la etapa final de alta tensión, concretamente el level shifter de 5 V a 32 V y el buffer de salida de alta tensión, ya se encontraban previamente implementados dentro del grupo de trabajo por el Dr. Iván Zamora. Por este motivo, en este capítulo se consideran como bloques disponibles para la integración del canal completo, pero no se desarrolla su layout interno como parte del trabajo realizado.

5.1.4. Dimensiones físicas de los bloques

La Tabla 5.1 resume las dimensiones físicas de los principales layouts empleados en la implementación. Todas las dimensiones se indican como ancho por alto y están expresadas en micrómetros.

Tabla 5.1: Dimensiones físicas de los principales bloques implementados.

Bloque	Ancho [μm]	Alto [μm]
Buffer CMOS	3.935	7.980
Flip-flop tipo D	18.055	8.700
Bloque IN	10.410	8.630
Registro serie-paralelo	178.590	16.065
Lógica de selección del DTC	43.625	14.030
Multiplexor 2:1	4.020	7.645
Multiplexor 8:1	21.140	10.050
Level shifter 1.5 V a 5 V	29.250	28.710
DTC	142.550	50.355
Canal completo	412.795	100.210
Sistema completo de ocho canales	1169.675	756.660

Estas dimensiones permiten comparar la contribución de cada bloque al área total. Como era esperable, los bloques repetitivos y de integración, como el DTC, el canal completo y el sistema multicanal, son los que ocupan mayor superficie. En cambio, celdas elementales como el buffer o el multiplexor 2:1 presentan áreas mucho más reducidas.

5.2. Layouts de bloques y celdas implementadas

5.2.1. Layout del buffer CMOS

La Figura 5.1 muestra el layout del buffer CMOS. Esta celda está formada por dos inversores conectados en cascada, tal como se describió en el apartado de implementación transistor-level. La señal de entrada atraviesa una primera etapa inversora y posteriormente una segunda etapa inversora, recuperando así la polaridad original en la salida.

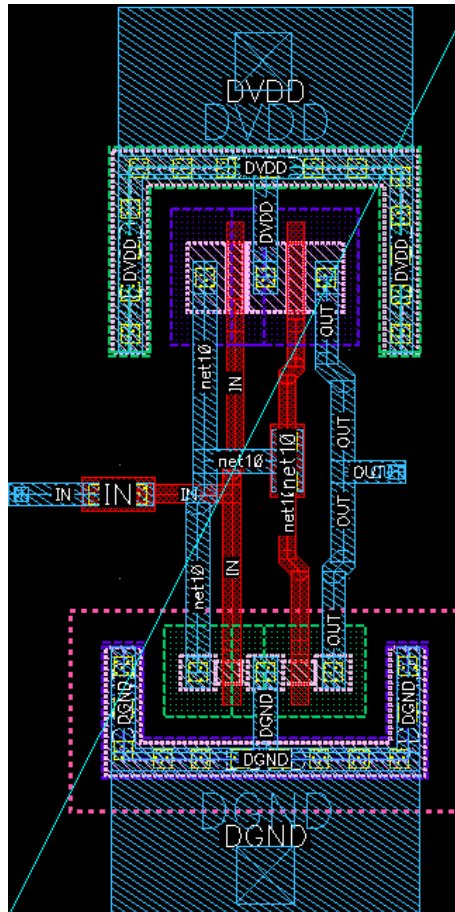


Figura 5.1: Layout del buffer CMOS implementado mediante dos inversores en cascada.

El layout del buffer ocupa una superficie de $3.935 \mu\text{m} \times 7.980 \mu\text{m}$. En la imagen se observa una disposición compacta, con el raíl de alimentación DVDD en la zona superior y el raíl de masa DGND en la zona inferior. La entrada IN se aplica a la primera etapa, mientras que la salida OUT se obtiene tras la segunda etapa.

Este bloque se utiliza para reforzar señales y mejorar la capacidad de conducción hacia etapas posteriores. En algunos puntos del diseño puede aparecer como celda individual, mientras que en otros casos su función se integra directamente dentro de bloques mayores para reducir área y evitar rutas innecesariamente largas.

5.2.2. Layout del flip-flop tipo D

La Figura 5.2 muestra el layout del flip-flop tipo D utilizado en la cadena de retardo. Esta celda secuencial recibe las señales D, CLK y RST, y genera las salidas Q y Q_N. Su estructura interna se basa en puertas lógicas con realimentaciones que permiten almacenar el estado entre flancos de reloj.

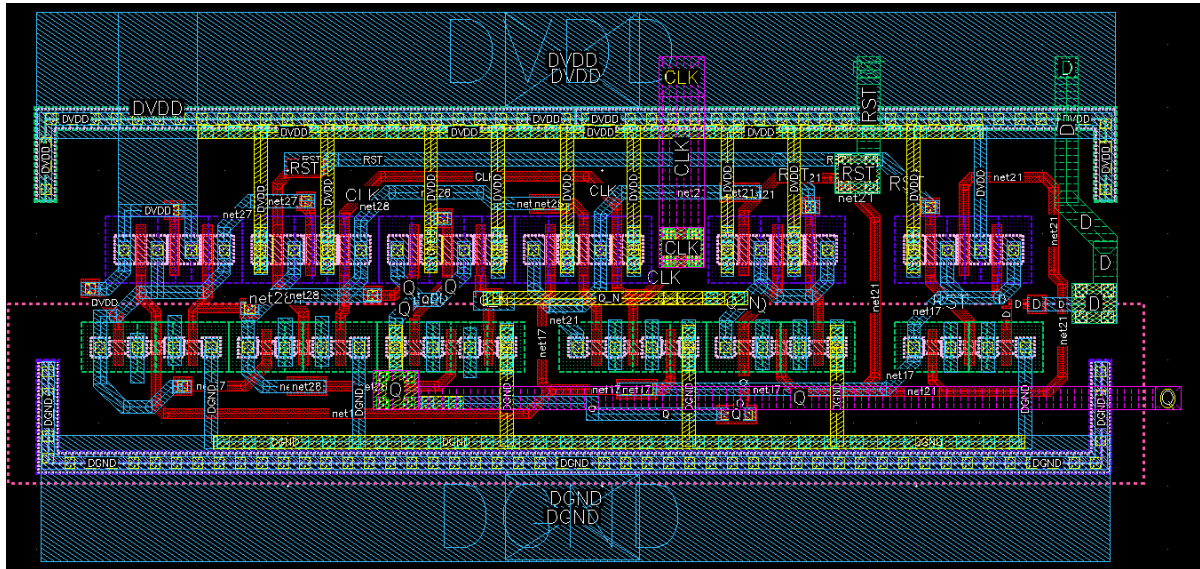


Figura 5.2: Layout del flip-flop tipo D con reset asíncrono activo bajo.

El flip-flop presenta unas dimensiones de $18.055 \mu\text{m} \times 8.700 \mu\text{m}$. Es el caso más claro de reutilización física dentro del diseño. A diferencia de otros bloques combinacionales, esta celda sí se ha diseñado una vez y posteriormente se ha copiado y colocado en las posiciones necesarias. Esto se debe a que tanto el registro serie-paralelo como la cadena DTC requieren varias etapas secuenciales equivalentes.

Esta forma de trabajar permite mantener una geometría regular, simplificar la distribución de CLK y RST, y evitar rediseñar manualmente la misma estructura varias veces. Además, al reutilizar la misma celda, el comportamiento físico de las distintas etapas resulta más homogéneo.

En el contexto del DTC, cada flip-flop constituye una etapa elemental de retardo. Al conectar varios flip-flops en serie, cada paso temporal habilitado introduce un retardo asociado a un periodo de reloj. Además, la entrada de reset de cada flip-flop permite activar o desactivar la etapa correspondiente mediante los bits de configuración.

5.2.3. Layout del bloque IN

El bloque IN es el encargado de generar las señales internas de operación a partir de las entradas externas DATA_IN, CONF_EN y CLK. Funcionalmente, este bloque habilita o deshabilita el reloj operativo durante la fase de configuración y controla el paso de la señal de entrada hacia el bloque multicanal.

La Figura 5.3 muestra el layout correspondiente al bloque IN.

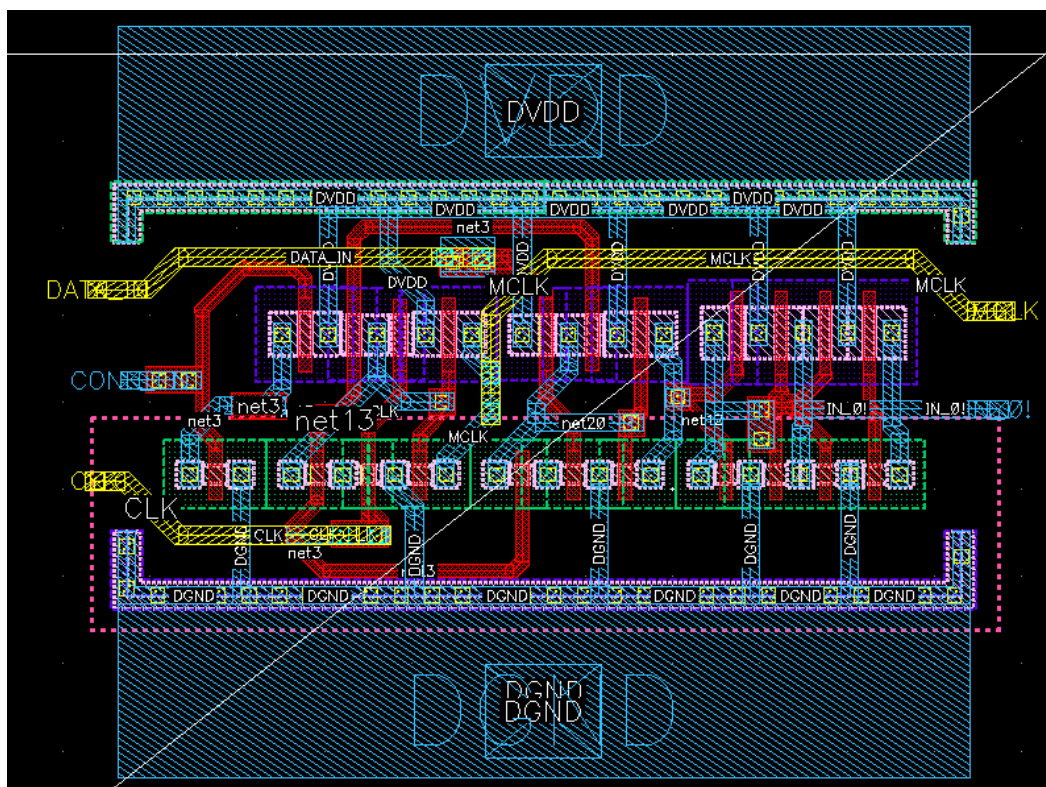


Figura 5.3: Layout del bloque IN, encargado de la habilitación de señales de entrada y reloj.

El bloque IN ocupa $10.410 \mu\text{m} \times 8.630 \mu\text{m}$. En la imagen se distinguen las señales principales del bloque, entre ellas DATA_IN, CONF_EN, CLK, MCLK e IN_0!. La organización física mantiene la alimentación DVDD en la parte superior y la referencia DGND en la parte inferior.

Este bloque se ha implementado como un módulo completo, no como una simple unión posterior de pequeñas puertas colocadas por separado. La lógica interna necesaria para generar MCLK e IN_0! se ha dispuesto de forma conjunta, buscando una implementación más compacta y reduciendo la longitud de las conexiones internas.

Durante configuración, CONF_EN mantiene deshabilitado el reloj operativo. Durante transmisión, MCLK queda habilitado y la señal de entrada puede propagarse hacia los canales. Esta separación entre configuración y operación queda reflejada tanto en el esquemático como en la organización física del bloque.

5.2.4. Layout del registro serie-paralelo

El registro serie-paralelo es el bloque encargado de cargar la palabra de configuración desde DATA_IN y entregar las señales internas D0–D7. Su implementación física se basa en la repetición de celdas secuenciales conectadas en cadena, lo que permite desplazar los bits de configuración con cada flanco útil del reloj de configuración.

La Figura 5.4 muestra el layout del registro serie-paralelo.

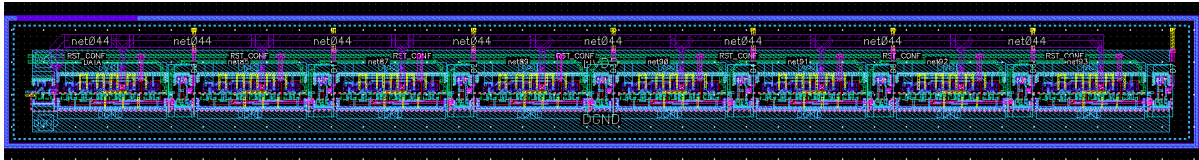


Figura 5.4: Layout del registro serie-paralelo de configuración.

El registro serie-paralelo tiene unas dimensiones de $178.590 \mu\text{m} \times 16.065 \mu\text{m}$. En este caso, el carácter repetitivo del bloque hace que la reutilización de la celda de flip-flop sea especialmente adecuada. Las distintas etapas del registro se colocan de forma alineada, compartiendo criterios de alimentación y masa. La señal de reloj de configuración y la señal de reset se distribuyen a lo largo del bloque, mientras que las salidas paralelas quedan disponibles para alimentar el DTC, la lógica de selección y el control de dirección.

Esta disposición permite obtener un bloque compacto y ordenado. Además, facilita la relación entre la descripción funcional del registro y su implementación física, ya que cada etapa del registro corresponde a un bit de configuración.

5.2.5. Layout del DTC

El DTC es el bloque encargado de generar los retardos programables. A nivel físico, está formado por una cadena de flip-flops, una serie de taps intermedios y una lógica de selección que determina qué versión retardada de la señal se entrega a la salida.

La Figura 5.5 muestra el layout de la cadena principal del DTC.

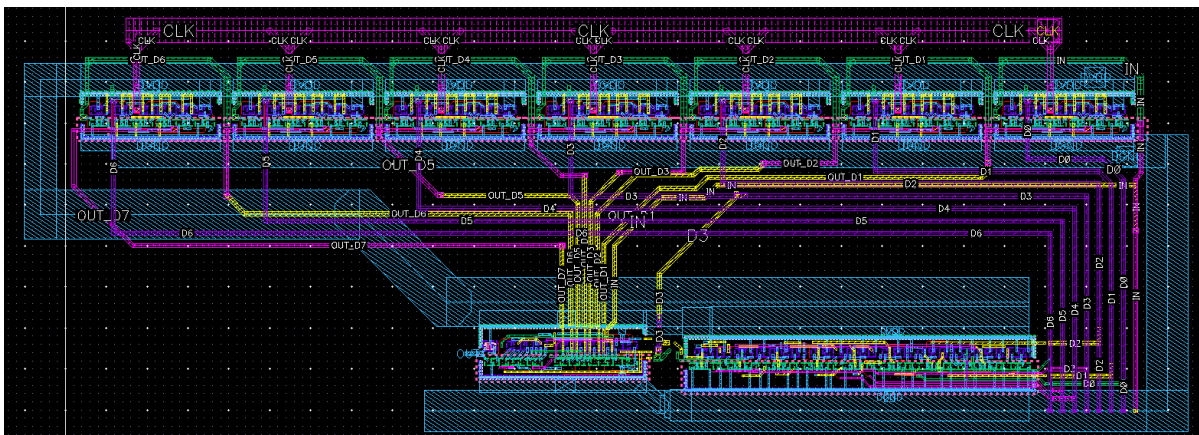


Figura 5.5: Layout de la cadena principal del DTC.

El bloque DTC ocupa $142.550 \mu\text{m} \times 50.355 \mu\text{m}$. En esta figura se aprecia la repetición de etapas secuenciales asociadas a los flip-flops del DTC. Esta parte del circuito se beneficia claramente de copiar y colocar la celda de flip-flop previamente diseñada, ya que la cadena requiere varias etapas similares conectadas de forma regular.

La señal de reloj operativo MCLK se distribuye hacia las diferentes etapas, mientras que las señales de configuración controlan el reset de cada flip-flop. Las salidas intermedias de la cadena

se extraen como taps, que posteriormente son utilizados por el multiplexor de selección.

La organización física del DTC responde directamente a su funcionamiento lógico. La señal de entrada se propaga de una etapa a la siguiente, generando distintas versiones retardadas. La selección de una u otra versión permite obtener el retardo programado.

5.2.6. Layout de la lógica de selección del DTC

Además de la cadena de flip-flops, el DTC requiere una lógica combinacional que genera las señales de selección del multiplexor. Esta lógica transforma la palabra de configuración en las señales de control necesarias para seleccionar el tap correspondiente.

La Figura 5.6 muestra el layout de la lógica de selección asociada al DTC.

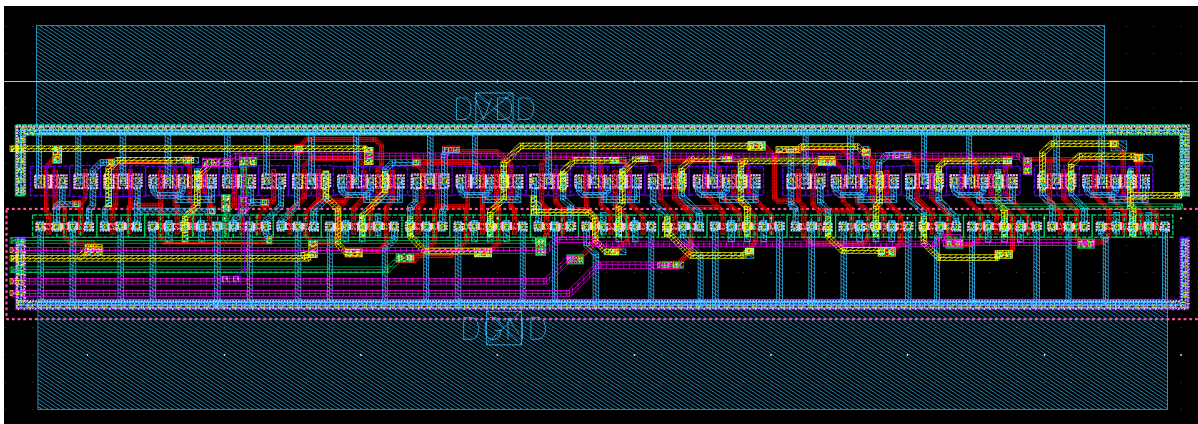


Figura 5.6: Layout de la lógica de selección del DTC.

La lógica de selección del DTC presenta unas dimensiones de $43.625\ \mu\text{m} \times 14.030\ \mu\text{m}$. Este bloque es uno de los ejemplos más representativos de la estrategia seguida en gran parte del layout. Aunque funcionalmente la lógica de selección puede explicarse mediante inversores, puertas XOR, puertas AND/NAND y otras conexiones internas, físicamente no se ha construido colocando cada puerta como una celda aislada para después unir las. En lugar de ello, se ha diseñado el módulo completo de forma conjunta.

Esta forma de trabajar permite compactar mucho más el layout, reducir las distancias entre señales relacionadas y evitar zonas vacías entre celdas independientes. Además, al tener todas las señales de configuración dentro del mismo bloque, resulta más sencillo optimizar el recorrido de las interconexiones internas.

La contrapartida es que el layout queda menos modular desde el punto de vista de reutilización de celdas individuales, pero en este caso el beneficio en área y compactación resulta más importante. Esta decisión es coherente con el objetivo del diseño, ya que el circuito final debe integrarse en una arquitectura multicanal donde el área disponible es un recurso crítico.

5.2.7. Layout del multiplexor 2:1

El multiplexor 2:1 se utiliza como bloque de selección. Su función es escoger una de dos señales de entrada en función de una señal de control. Este tipo de estructura se utiliza tanto en la selección de dirección de los canales como en la construcción de multiplexores de mayor número de entradas.

La Figura 5.7 muestra el layout del multiplexor 2:1.

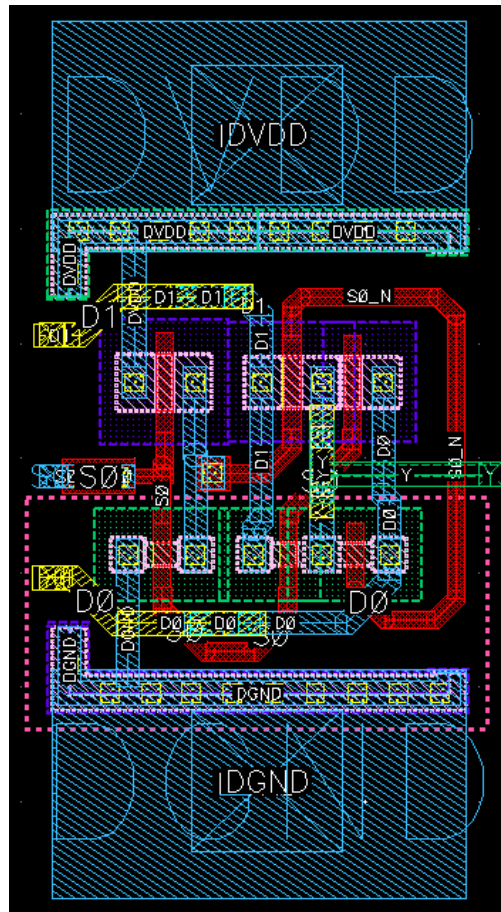


Figura 5.7: Layout del multiplexor 2:1.

El multiplexor 2:1 ocupa $4.020 \mu\text{m} \times 7.645 \mu\text{m}$. En el layout se observan las entradas de datos, la señal de selección y su complementaria, así como la salida del multiplexor. La celda se organiza de forma compacta, manteniendo los raíles de alimentación y masa en las zonas superior e inferior.

En algunos casos, este tipo de estructura puede reutilizarse como bloque elemental. Sin embargo, cuando el multiplexor forma parte de un módulo mayor, la colocación física puede ajustarse para favorecer una mejor compactación global, en lugar de mantener una separación rígida entre celdas individuales.

5.2.8. Layout del multiplexor 8:1

El multiplexor 8:1 es el bloque encargado de seleccionar uno de los taps del DTC. Sus entradas corresponden a las distintas versiones retardadas de la señal, y sus señales de selección determinan cuál de ellas se propaga hacia la salida.

La Figura 5.8 muestra el layout del multiplexor 8:1.

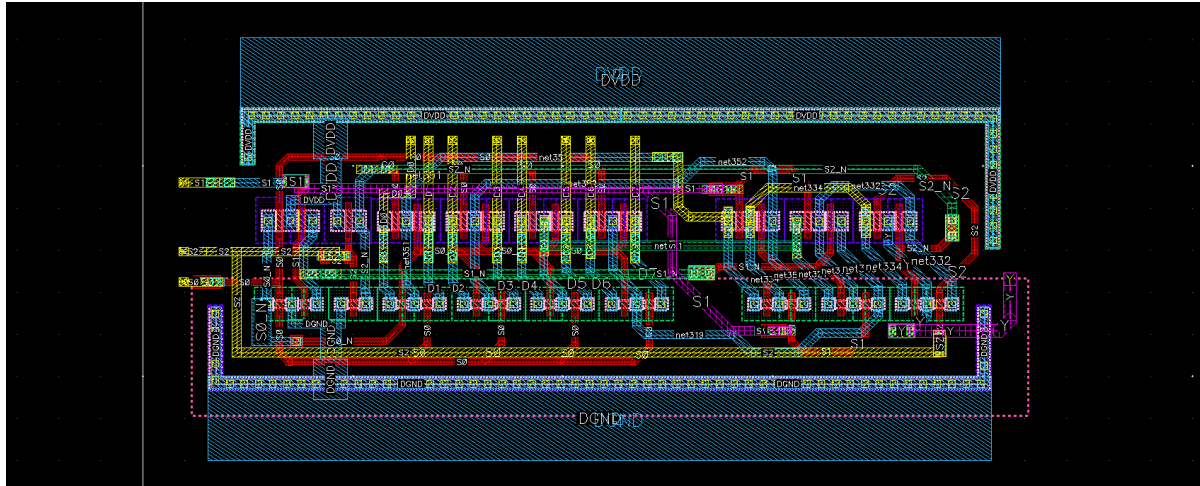


Figura 5.8: Layout del multiplexor 8:1 utilizado para la selección del tap retardado.

El multiplexor 8:1 presenta unas dimensiones de $21.140 \mu\text{m} \times 10.050 \mu\text{m}$. Este bloque presenta una mayor complejidad física debido al número de entradas y a la necesidad de distribuir las señales de selección. Las señales $S0$, $S1$ y $S2$, junto con sus versiones complementarias, permiten seleccionar una de las ocho entradas disponibles.

En este caso, la implementación física busca reducir la longitud de las rutas internas y mantener una distribución compacta de las entradas, señales de selección y salida. Al tratarse de un bloque con varias rutas de datos, un layout demasiado fragmentado aumentaría el área y complicaría el enrutado. Por este motivo, el bloque se trata como una estructura completa, optimizando la colocación interna de sus señales.

Desde el punto de vista funcional, este multiplexor es el encargado de convertir la cadena de taps del DTC en una única salida retardada. Por tanto, constituye el punto donde la configuración digital se traduce en la selección efectiva del retardo.

5.2.9. Layout del level shifter de 1.5 V a 5 V

Tras la lógica digital de baja tensión, la señal debe adaptarse a niveles superiores antes de atacar las etapas de alta tensión. Para ello se utiliza un bloque de adaptación de nivel de 1.5 V a 5 V. Este bloque permite conectar la lógica digital de control con las etapas posteriores que trabajan con una alimentación mayor.

La Figura 5.9 muestra el layout del level shifter.

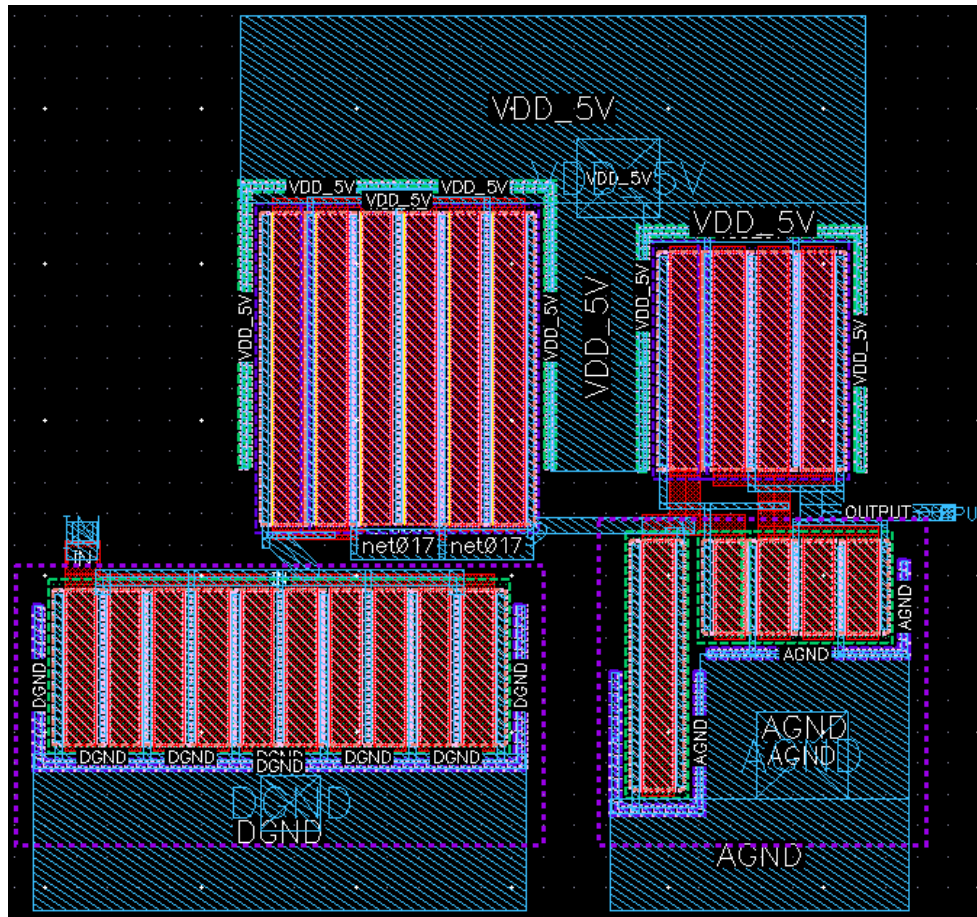


Figura 5.9: Layout del level shifter de 1.5 V a 5 V.

El level shifter de 1.5 V a 5 V ocupa $29.250 \mu\text{m} \times 28.710 \mu\text{m}$. En este layout se observa una separación más marcada entre dominios de alimentación. Aparecen referencias asociadas a VDD_5V y AGND, además de las conexiones necesarias para recibir la señal de entrada y generar la salida adaptada. Esta separación es importante porque el bloque actúa como interfaz entre la lógica digital de baja tensión y las etapas que requieren niveles de tensión superiores.

El level shifter constituye, por tanto, un bloque intermedio esencial dentro del canal de transmisión. Permite que las señales generadas por el DTC y la lógica de selección puedan propagarse hacia las etapas de salida manteniendo niveles eléctricos compatibles.

5.3. Integración física de bloques

A partir de los layouts anteriores se construyen los bloques funcionales de mayor nivel. Sin embargo, conviene remarcar que la integración física no se ha planteado siempre como una suma directa de celdas pequeñas previamente acabadas. En muchos casos, especialmente en la lógica combinacional, se ha trabajado directamente sobre el módulo completo para optimizar área y enrutado.

Por ejemplo, en bloques como la lógica del DTC, resulta más eficiente colocar conjuntamente los transistores y conexiones asociados a inversores, XOR, AND/NAND y señales de control, en lugar de insertar cada puerta como una celda independiente. Esta estrategia permite aprovechar mejor el espacio físico, compartir rutas cercanas y reducir la longitud de interconexiones.

En cambio, los flip-flops sí se han tratado como una celda repetitiva. Una vez diseñado su layout, se ha reutilizado mediante copia y colocación en el registro serie-paralelo y en la cadena DTC. Esta diferencia de criterio responde a la naturaleza de cada bloque: los flip-flops son estructuras repetidas y regulares, mientras que la lógica combinacional de selección se beneficia más de una implementación compacta a nivel de módulo.

Por tanto, la metodología seguida combina jerarquía funcional con optimización física. El diseño se entiende por bloques, pero no todos los bloques se construyen físicamente de la misma manera. Cuando una celda es repetitiva, se reutiliza. Cuando un módulo puede compactarse mejor diseñándolo como conjunto, se implementa directamente como bloque completo.

En la integración de los canales también se emplean bloques de alta tensión previamente disponibles en el grupo de investigación. En particular, el level shifter de 5 V a 32 V y el buffer de salida de alta tensión fueron implementados previamente por el Dr. Iván Zamora. Estos bloques se incorporan dentro del canal completo para permitir la generación de la señal final de 32 V, pero su diseño interno no se detalla en este capítulo al no formar parte de la implementación desarrollada específicamente en este trabajo.

5.3.1. Layout de un canal completo

Una vez diseñados los bloques digitales principales y disponiendo de las etapas de alta tensión necesarias, se realiza la integración física de un canal completo. Este canal incluye la cadena de retardo programable, la lógica de selección, los multiplexores, la adaptación de nivel de 1.5 V a 5 V, el level shifter de 5 V a 32 V y el buffer final de alta tensión.

La Figura 5.10 muestra el layout de un canal completo.

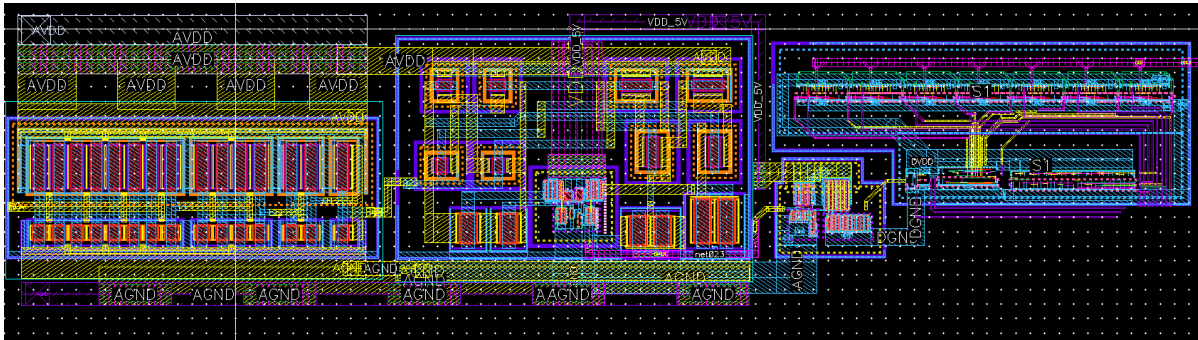


Figura 5.10: Layout de un canal completo de transmisión.

El layout de un canal completo ocupa $412.795 \mu\text{m} \times 100.210 \mu\text{m}$. Esta dimensión refleja la acumulación de los distintos bloques necesarios para generar una salida temporizada y elevada hasta 32 V. A diferencia de las celdas individuales, el canal completo no solo debe resolver la implementación interna de cada bloque, sino también la conexión entre dominios de tensión diferentes y el encaminamiento de señales de control, reloj y propagación.

Desde el punto de vista funcional, este layout representa la unidad mínima repetible del sistema multicanal. Cada canal recibe la configuración común, el reloj operativo y las señales de propagación, y genera una salida de transmisión asociada a uno de los elementos del array de PMUTs. Por tanto, la correcta integración física de este bloque es clave para construir posteriormente el sistema de ocho canales.

5.3.2. Layout del sistema completo de ocho canales

Finalmente, se integra el sistema completo de ocho canales. Esta implementación agrupa los canales de transmisión y las señales necesarias para controlar la propagación temporal entre ellos. La finalidad de este layout es comprobar que la arquitectura puede organizarse físicamente como un bloque multicanal completo, manteniendo la conexión entre las señales de configuración, reloj, entrada de burst y salidas de alta tensión.

La Figura 5.11 muestra el layout del sistema completo.

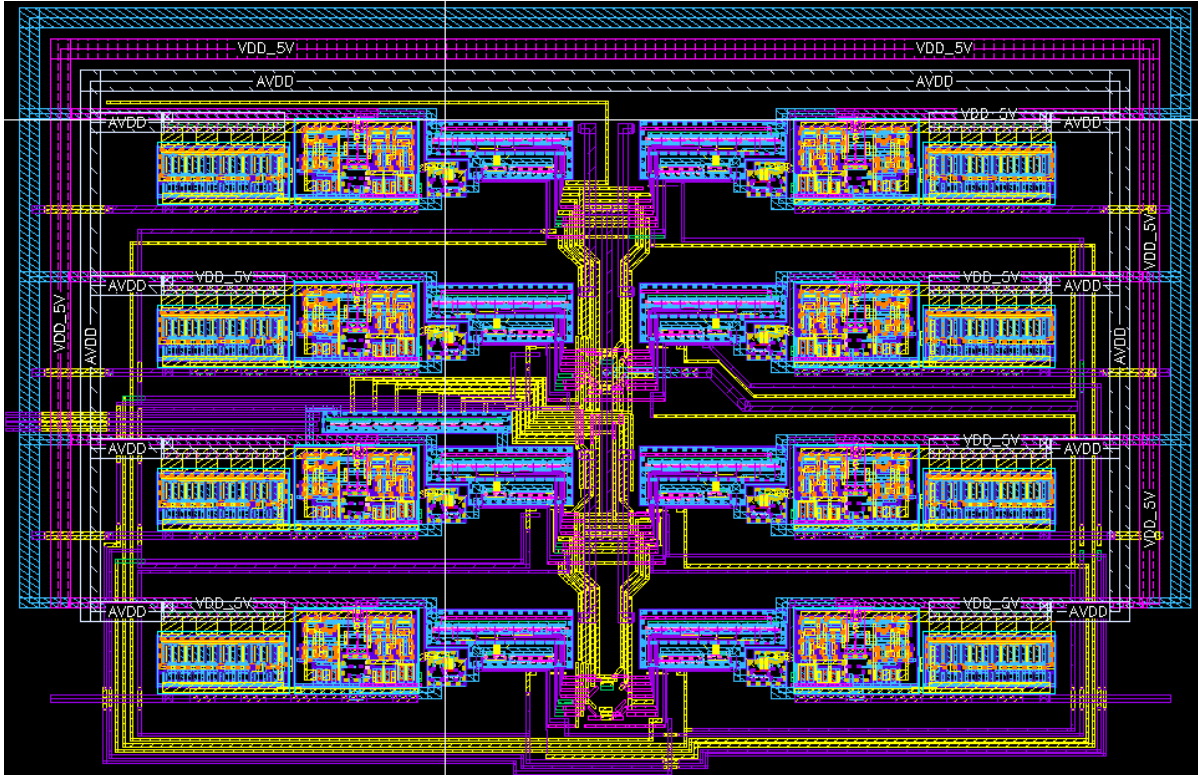


Figura 5.11: Layout del sistema completo de ocho canales.

El layout del sistema completo presenta unas dimensiones de $1169.675 \mu\text{m} \times 756.660 \mu\text{m}$. Esta superficie incluye la integración de los ocho canales y las interconexiones necesarias para distribuir las señales comunes. En comparación con un único canal, el sistema completo introduce una mayor complejidad de rutado debido a la repetición de bloques, la distribución del reloj operativo y la conexión de las rutas de propagación directa e inversa entre canales.

La implementación del sistema completo permite verificar la viabilidad física de la arquitectura propuesta. Además, proporciona una estimación realista del área ocupada por el bloque multicanal y muestra cómo los canales pueden organizarse de forma conjunta dentro del layout global.

5.4. Resumen

En este apartado se ha presentado la implementación física mediante layout de los principales bloques de la arquitectura. Se han mostrado celdas y bloques como el buffer, el flip-flop tipo D, el bloque IN, el registro serie-paralelo, el DTC, la lógica de selección, los multiplexores y el level shifter de 1.5 V a 5 V.

También se ha incluido la integración física de un canal completo y del sistema completo de ocho canales. El canal completo presenta unas dimensiones de $412.795\ \mu\text{m} \times 100.210\ \mu\text{m}$, mientras que el sistema completo ocupa $1169.675\ \mu\text{m} \times 756.660\ \mu\text{m}$. Estas medidas permiten valorar la superficie final necesaria para implementar la arquitectura de transmisión multicanal.

La implementación física se ha realizado siguiendo una estrategia mixta. En los bloques repetitivos, especialmente en los flip-flops, se ha diseñado una celda individual y se ha reutilizado mediante copia y colocación. Esto permite mantener regularidad y coherencia entre etapas equivalentes.

En cambio, en gran parte de la lógica combinacional no se ha seguido una metodología basada únicamente en colocar puertas sueltas y conectarlas después. En bloques como la lógica del DTC, el layout se ha realizado directamente a nivel de módulo, integrando las puertas y conexiones necesarias dentro de una misma estructura física. Esta decisión permite reducir área, compactar rutas internas y mejorar la eficiencia espacial del diseño.

Por último, se ha indicado que algunos bloques de alta tensión, como el level shifter de 5 V a 32 V y el buffer de salida de alta tensión, proceden de implementaciones previas realizadas por el Dr. Iván Zamora. En este trabajo se han utilizado como bloques de integración dentro del canal completo, centrando la documentación en los bloques digitales, la adaptación de 1.5 V a 5 V, el canal completo y la integración del sistema de ocho canales.

Capítulo 6

Validación funcional a nivel de sistema

Una vez definida la arquitectura, la implementación a nivel de transistor y el layout de los bloques principales, se realiza una validación funcional del sistema completo. El objetivo de este capítulo no es repetir la caracterización individual de las puertas lógicas, sino comprobar que los bloques trabajan de forma coordinada y que la arquitectura genera la secuencia temporal esperada en los ocho canales de transmisión.

La validación se plantea a nivel funcional y se complementa con simulaciones transitorias realizadas en Cadence Virtuoso. Por tanto, se centra en la relación temporal entre las señales principales: carga de la palabra de programación, habilitación del reloj operativo, aplicación de los pulsos de transmisión, generación del retardo programado y activación ordenada de las salidas multicanal. Esta comprobación complementa las simulaciones DC y temporales del Capítulo 4 y la implementación física descrita en el Capítulo 5.

En este capítulo no se vuelve a representar la arquitectura general completa, ya que dicha vista se ha presentado previamente. En su lugar, se analiza si el comportamiento temporal que se espera de esa arquitectura se cumple de forma coherente con las señales de programación D0–D7 y con el reloj externo de 100 MHz.

Con este reloj, el periodo utilizado como paso temporal del sistema es:

$$T_{clk} = \frac{1}{100 \text{ MHz}} = 10 \text{ ns}$$

Por tanto, para un número de pasos temporales n , el retardo programado queda:

$$\Delta t_n = nT_{clk}$$

Esta expresión resume la función del DTC dentro del sistema: convertir una palabra de programación en una separación temporal discreta entre canales.

6.1. Secuencia funcional validada

El sistema trabaja en dos fases. En la fase de programación, `CONF_EN` se encuentra activa y `DATA_IN` transporta los bits que se almacenan en `CONF_LOGIC`. Durante esta fase, los canales no deben operar, por lo que el reloj operativo `MCLK` permanece deshabilitado.

La relación funcional utilizada para el reloj operativo es:

$$\text{MCLK} = \text{CLK} \cdot \overline{\text{CONF_EN}}$$

De esta forma, cuando `CONF_EN = 1`, se obtiene:

$$\text{MCLK} = 0$$

y los canales no reciben flancos de operación. Cuando la programación termina, `CONF_EN` pasa a nivel bajo, los bits D0–D7 quedan fijados y `MCLK` queda habilitado como reloj operativo del bloque de transmisión:

$$\text{CONF_EN} = 0 \Rightarrow \text{MCLK} = \text{CLK}$$

A partir de ese momento, `DATA_IN` deja de interpretarse como palabra de programación y pasa a transportar los pulsos de transmisión. Esta separación es importante porque una misma entrada externa cumple dos funciones distintas según el estado de `CONF_EN`.

6.2. Validación del retardo programable

La primera comprobación funcional corresponde al DTC. Este bloque debe entregar una versión retardada de la señal de entrada, con un desplazamiento temporal definido por el número de pasos temporales seleccionados.

Para cada valor de n , el retardo esperado viene dado por:

$$\Delta t_n = nT_{clk}$$

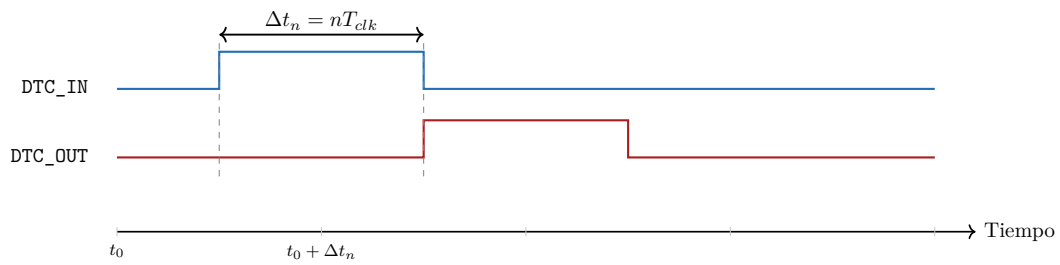
La Tabla 6.1 resume los retardos asociados a todos los valores disponibles de n para $T_{clk} = 10$ ns.

La Figura 6.1 representa la comprobación funcional del DTC de forma general. La salida conserva la forma temporal del pulso de entrada, pero aparece desplazada una cantidad Δt_n determinada por la programación cargada.

Esta comprobación permite verificar que la cadena de retardo se comporta como un selector temporal de taps. El valor de n no se interpreta como un número binario convencional, sino

Tabla 6.1: Retardos programables del DTC para $f_{clk} = 100$ MHz.

n	$\Delta t_n = nT_{clk}$	Retardo [ns]
0	$0T_{clk}$	0
1	$1T_{clk}$	10
2	$2T_{clk}$	20
3	$3T_{clk}$	30
4	$4T_{clk}$	40
5	$5T_{clk}$	50
6	$6T_{clk}$	60
7	$7T_{clk}$	70

**Figura 6.1:** Validación funcional general del retardo introducido por el DTC.

como el número de pasos temporales aplicados sobre la señal.

6.3. Validación multicanal y selección de sentido

Una vez validado el retardo de un canal, se comprueba la secuencia temporal de los ocho canales. En dirección directa, la activación comienza en CH01 y avanza hasta CH08. Si el primer canal se activa en el instante t_0 , el instante de activación de cada canal viene dado por:

$$t_{CHk} = t_0 + (k - 1)\Delta t_n$$

con $k = 1, 2, \dots, 8$. En dirección inversa, el orden temporal se invierte y la secuencia comienza en CH08. La Tabla 6.2 resume ambos sentidos de propagación.

Tabla 6.2: Instantes de activación esperados en función del sentido de propagación.

Canal	Dirección directa, D7=0	Dirección inversa, D7=1
CH01	t_0	$t_0 + 7\Delta t_n$
CH02	$t_0 + \Delta t_n$	$t_0 + 6\Delta t_n$
CH03	$t_0 + 2\Delta t_n$	$t_0 + 5\Delta t_n$
CH04	$t_0 + 3\Delta t_n$	$t_0 + 4\Delta t_n$
CH05	$t_0 + 4\Delta t_n$	$t_0 + 3\Delta t_n$
CH06	$t_0 + 5\Delta t_n$	$t_0 + 2\Delta t_n$
CH07	$t_0 + 6\Delta t_n$	$t_0 + \Delta t_n$
CH08	$t_0 + 7\Delta t_n$	t_0

La Figura 6.2 muestra la misma idea de forma gráfica. Se representa una separación constante Δt_n entre canales consecutivos y se observa cómo D7 invierte el orden de activación sin cambiar la magnitud del retardo.

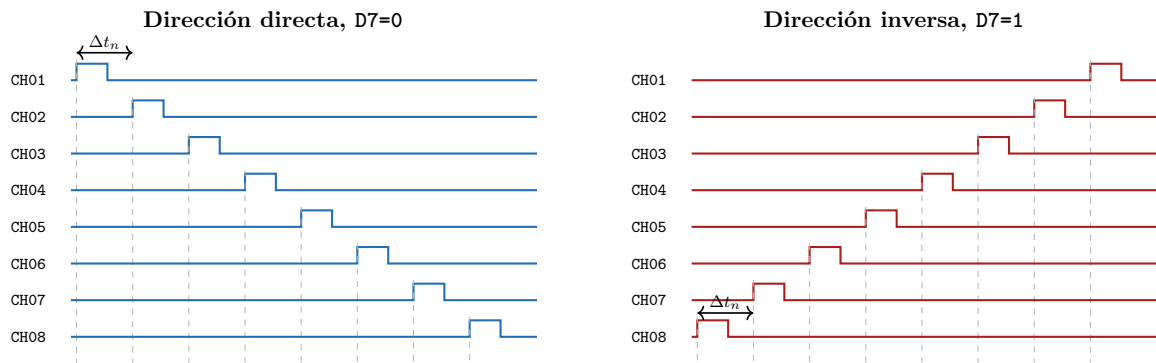


Figura 6.2: Validación funcional del orden de activación de los ocho canales para ambos sentidos de propagación.

El resultado funcional esperado es que D7 actúe únicamente como bit de sentido. El retardo entre canales consecutivos sigue siendo Δt_n , pero la secuencia se aplica de CH01 a CH08 o de CH08 a CH01 según el valor de este bit.

6.4. Simulación transitoria de las salidas multicanal

Para complementar la validación conceptual anterior, se realizaron simulaciones transitorias en Cadence Virtuoso de la respuesta temporal del sistema durante la fase de transmisión. En estas simulaciones se empleó un reloj externo de $f_{clk} = 100$ MHz, por lo que el periodo de reloj es $T_{clk} = 10$ ns. Una vez finalizada la fase de programación, la misma señal DATA_IN transporta tres pulsos de transmisión a 20 MHz, iniciados alrededor de 110 ns. Las salidas representadas corresponden al dominio de alta tensión, con pulsos de aproximadamente 32 V.

En primer lugar se validó el caso de propagación directa, con D7 = 0. La palabra de configuración aplicada mediante DATA_IN es 01100000. Para esta configuración se seleccionaron dos pasos temporales de retardo entre canales, activando D0 y D1. Por tanto, el retardo esperado entre canales consecutivos es:

$$\Delta t = 2T_{clk} = 20 \text{ ns}$$

La Figura 6.3 muestra la simulación del sistema completo para esta configuración. Se observa que la secuencia de activación comienza en CH01 y se propaga progresivamente hasta CH08, manteniendo la forma temporal de los pulsos de transmisión y la separación temporal esperada entre canales.

A continuación se validó el comportamiento en sentido inverso, configurando D7 = 1. La palabra de configuración aplicada mediante DATA_IN es 11111000. En este caso se seleccionaron cuatro pasos temporales de retardo entre canales, por lo que:

$$\Delta t = 4T_{clk} = 40 \text{ ns}$$

La Figura 6.4 muestra la simulación correspondiente. En este caso, la secuencia temporal se invierte: el primer canal en activarse es CH08 y la propagación avanza en sentido contrario hasta CH01.

Las dos simulaciones permiten comprobar el comportamiento funcional de la arquitectura completa en dos configuraciones representativas. En ambos casos se mantiene la temporización esperada entre canales, la forma temporal de los pulsos de transmisión y el nivel final de salida de 32 V, confirmando la operación conjunta de la lógica de configuración, la cadena de retardos, la selección de sentido y las etapas de salida.

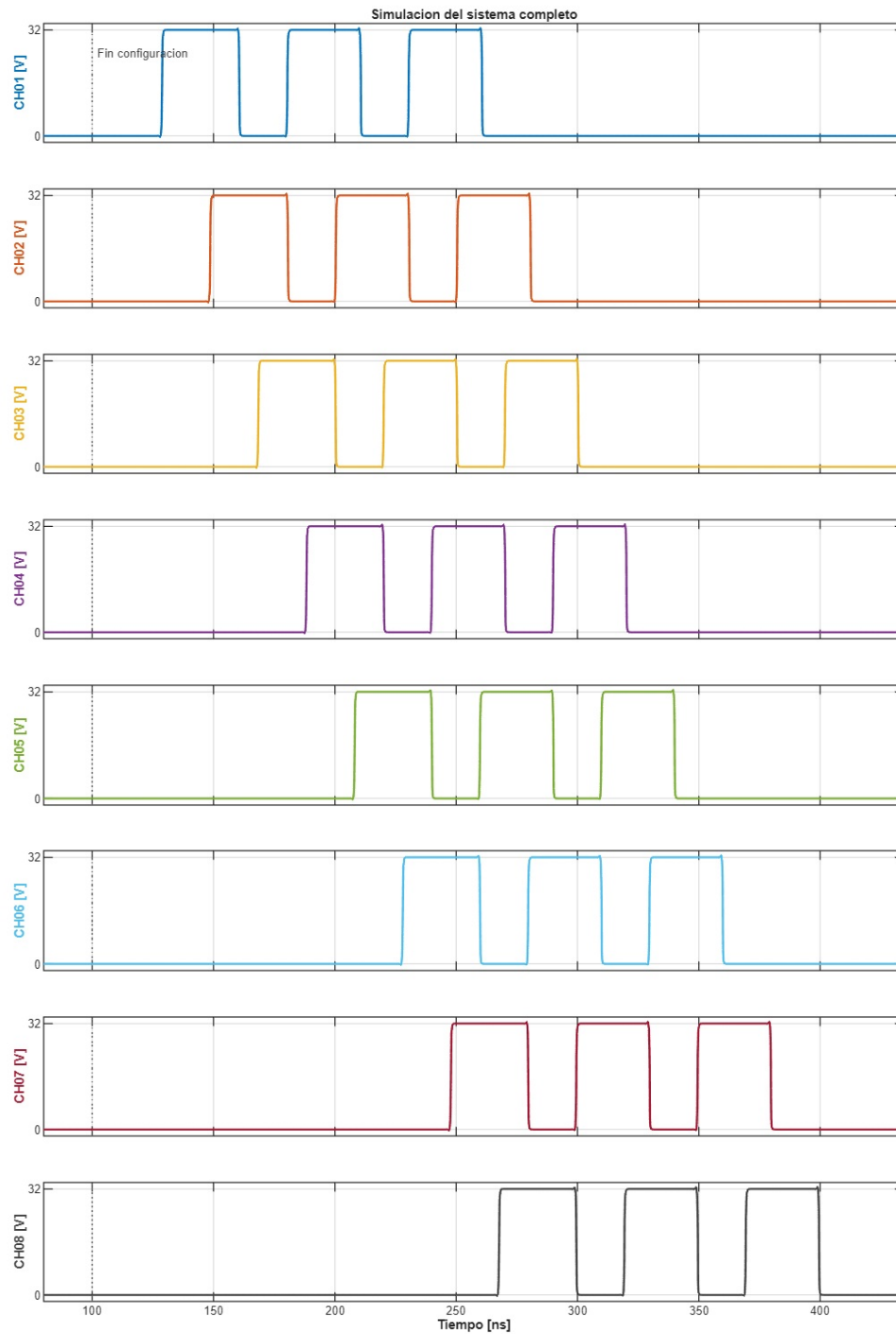


Figura 6.3: Simulación transitoria del sistema completo en dirección directa. La configuración aplicada mediante DATA_IN es 01100000, con $D7 = 0$, $f_{clk} = 100$ MHz, retardo de $2T_{clk} = 20$ ns entre canales y tres pulsos de transmisión a 20 MHz iniciados alrededor de 110 ns.

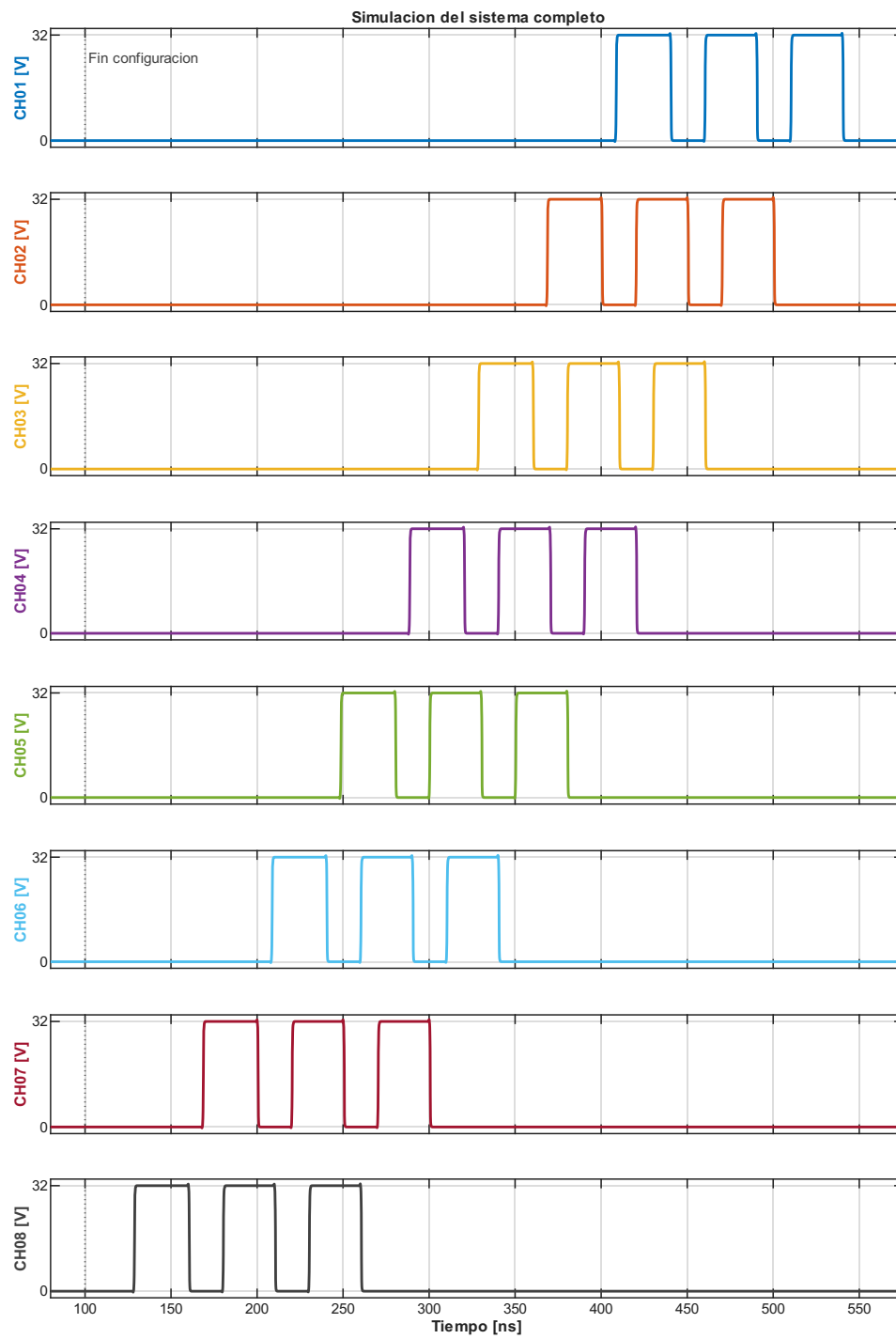


Figura 6.4: Simulación transitoria del sistema completo en dirección inversa. La configuración aplicada mediante DATA_IN es 11111000, con $D7 = 1$, $f_{clk} = 100$ MHz, retardo de $4T_{clk} = 40$ ns entre canales y tres pulsos de transmisión a 20 MHz iniciados alrededor de 110 ns.

6.5. Relación con el ángulo de barrido

La validación temporal anterior se conecta directamente con el objetivo físico del sistema. Para una separación entre elementos p , el retardo entre canales consecutivos se relaciona con el ángulo de direccionamiento mediante:

$$\Delta t = \frac{p \sin(\theta)}{c}$$

Si se adopta el criterio $p = \lambda/2$, con $\lambda = c/f$, se obtiene:

$$\theta_n = \arcsin(2f\Delta t_n)$$

Como el retardo programado por la electrónica es:

$$\Delta t_n = nT_{clk}$$

la expresión final queda:

$$\theta_n = \arcsin(2fnT_{clk})$$

Por tanto, el capítulo no valida una única dirección concreta, sino el mecanismo completo que permite generar el conjunto de direcciones discretas: el valor de n define el módulo del retardo, y D7 determina el sentido de aplicación de la secuencia temporal.

6.6. Representación funcional de la adaptación de nivel

Además de la temporización, el sistema debe adaptar las señales digitales generadas a baja tensión hasta el nivel necesario para excitar los PMUTs. A nivel funcional, el recorrido de cada canal puede resumirse en tres dominios:

$$1.5\text{ V} \rightarrow 5\text{ V} \rightarrow 32\text{ V}$$

La Figura 6.5 muestra esta adaptación de forma esquemática. Se ha representado la misma ventana temporal en los tres dominios para remarcar que la información temporal del burst se conserva; lo que cambia es el nivel eléctrico de la señal.

Esta representación no sustituye a la simulación eléctrica de los level shifters, sino que resume su papel dentro del sistema completo: transportar el mismo pulso temporal desde el dominio digital hasta la salida de alta tensión.

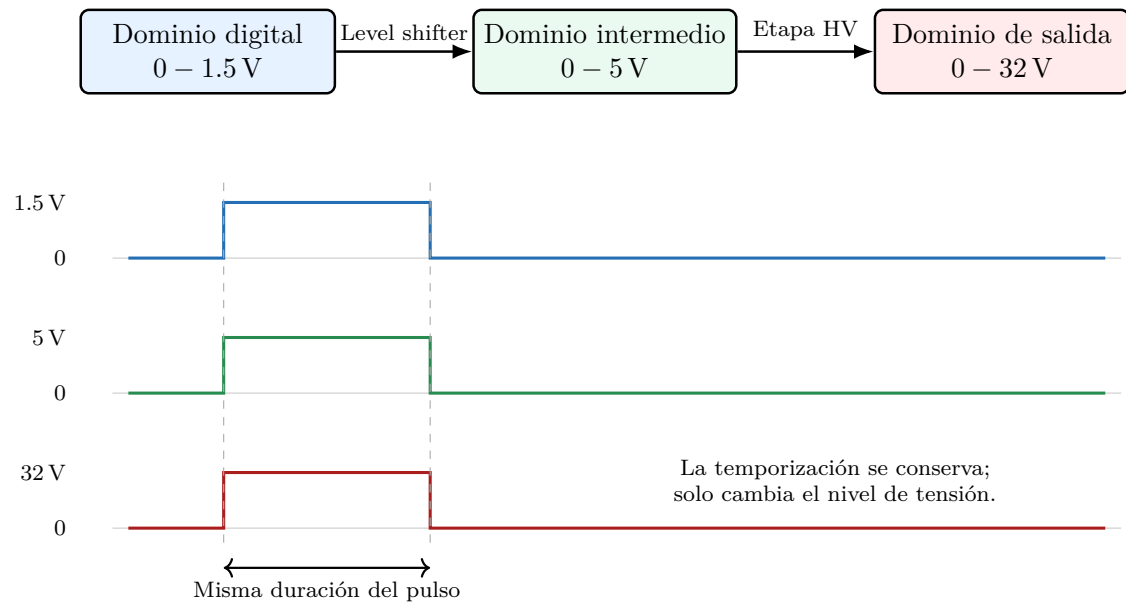


Figura 6.5: Representación funcional de la adaptación de nivel desde la lógica digital hasta la salida de alta tensión.

6.7. Resumen de validaciones funcionales

La Tabla 6.3 resume las comprobaciones realizadas a nivel de sistema.

Tabla 6.3: Resumen de validaciones funcionales a nivel de sistema.

Comprobación	Objetivo	Resultado esperado
Programación/transmisión	Verificar que el sistema separa la carga de bits y el envío de pulsos de transmisión	MCLK deshabilitado durante programación y habilitado durante transmisión
DTC	Comprobar la generación de retardos discretos	$\Delta t_n = nT_{clk}$ para $n = 0, \dots, 7$
Ocho canales, dirección directa	Verificar el orden CH01 a CH08	Activación progresiva con separación Δt_n
Ocho canales, dirección inversa	Verificar el efecto de D7	Activación progresiva en orden inverso
Simulación transitoria, D7 = 0	Validar la propagación directa con señales de salida de 32 V	Secuencia CH01 a CH08 con retardo próximo a $2T_{clk}$
Simulación transitoria, D7 = 1	Validar la propagación inversa con señales de salida de 32 V	Secuencia CH08 a CH01 con retardo de $4T_{clk}$
Relación con beamforming	Conectar retardo electrónico y dirección acústica	$\theta_n = \arcsin(2fnT_{clk})$
Adaptación de nivel	Comprobar el papel funcional de las etapas de salida	Misma temporización en los dominios de 1.5 V, 5 V y 32 V

6.8. Alcance de la validación funcional

La validación presentada en este capítulo se centra en la secuencia lógica y temporal del sistema. Su objetivo es comprobar que la arquitectura completa responde de forma coherente desde el punto de vista funcional: carga de programación, habilitación del reloj operativo, generación del retardo, selección del sentido y obtención de salidas temporizadas.

Quedan fuera de esta validación algunos efectos físicos propios de una caracterización post-layout completa, como capacidades parásitas de interconexión, resistencias de metal y contactos, desajustes entre etapas repetidas, variaciones de proceso y temperatura, carga real de las etapas de alta tensión y diferencias de retardo entre canales debidas al layout.

Esto no invalida la simulación funcional, sino que delimita su alcance. En esta fase se comprueba que la arquitectura trabaja correctamente como sistema. La caracterización eléctrica fina de las celdas y el desarrollo físico de los bloques se han tratado en los capítulos anteriores.

Capítulo 7

Conclusiones

En este Trabajo Fin de Grado se ha desarrollado y documentado una arquitectura digital orientada a la generación de retardos programables para aplicaciones de *beamforming* ultrasónico mediante tecnología CMOS HV. El objetivo principal era controlar la temporización relativa entre varios canales de transmisión, de forma que la excitación de un array de PMUTs pudiera realizarse con retardos discretos y programables.

A lo largo de la memoria se ha partido de la base física del problema, relacionando los retardos temporales entre elementos de un array con el ángulo de direccionamiento del haz. Esta relación ha permitido justificar la necesidad de una cadena digital de retardos y de una lógica de configuración capaz de seleccionar tanto la magnitud del retardo como el sentido de propagación entre canales.

Desde el punto de vista arquitectónico, el sistema se ha organizado en tres bloques principales: el bloque de entrada y gestión de señales, la lógica de configuración serie-paralelo y el bloque de transmisión de ocho canales. Esta separación permite distinguir claramente la fase de programación, en la que se cargan los bits de control, y la fase de transmisión, en la que los bursts se aplican a los canales utilizando la configuración almacenada.

La arquitectura propuesta permite seleccionar hasta siete pasos temporales de retardo, además del caso central sin retardo. Considerando ambos sentidos de propagación mediante el bit D7, el sistema proporciona un conjunto de posiciones discretas de barrido. Para el caso de referencia de un PMUT de 5 MHz y un reloj de 100 MHz, se obtiene un rango angular aproximado de -44.43° a $+44.43^\circ$, lo que resulta coherente con el objetivo de disponer de una arquitectura digital útil para direccionamiento ultrasónico.

A nivel de circuito, se han implementado y caracterizado las celdas digitales básicas necesarias para construir la arquitectura: inversores, buffers, puertas lógicas, multiplexores, flip-flops y etapas de adaptación de nivel. Las simulaciones DC permiten identificar el punto de conmutación de las celdas, definido por la condición $V_{in} = V_{out}$, mientras que las respuestas temporales con una carga de 100 fF permiten estimar retardos de propagación y tiempos de subida y bajada.

También se ha desarrollado la implementación física mediante layout de los bloques principales.

En esta fase se ha trabajado con la tecnología CMOS HV de SilTerra de $0.13\mu\text{m}$, utilizando Cadence Virtuoso para el diseño físico y herramientas de verificación como Calibre para DRC, LVS y extracción. La metodología seguida combina reutilización de celdas repetitivas, como los flip-flops, con implementación compacta de bloques combinacionales cuando resulta más conveniente desde el punto de vista de área y enrutado.

La validación funcional a nivel de sistema confirma que la arquitectura realiza correctamente la secuencia esperada: carga de programación, habilitación del reloj operativo, generación de retardos discretos, selección del sentido de propagación y obtención de salidas temporizadas en los ocho canales. Esta validación no sustituye a una caracterización post-layout completa, pero sí permite comprobar que la estructura propuesta funciona de forma coherente como sistema.

Como trabajo futuro, sería conveniente completar la caracterización eléctrica post-layout del sistema integrado, incluyendo las capacidades parásitas, las diferencias de retardo entre canales y el efecto de la carga real de las etapas de alta tensión. También resultaría interesante ampliar la validación hacia una simulación más cercana al comportamiento acústico del array, conectando las salidas temporizadas del circuito con un modelo de emisión de los PMUTs. De esta forma se podría cerrar el ciclo completo entre electrónica de control, excitación de los transductores y formación del haz ultrasónico.

En conjunto, el trabajo realizado demuestra la viabilidad de una solución digital para controlar retardos temporales en un sistema ultrasónico multicanal. La arquitectura desarrollada constituye una base sólida para futuras etapas de integración, caracterización y validación experimental dentro del contexto de sistemas ultrasónicos integrados en tecnología CMOS HV.

Bibliografía

- [1] N. Setter, ed. *Electroceramic-Based MEMS: Fabrication-Technology and Applications*. 1st ed. Boston: Springer, 2005. ISBN: 978-0-387-23319-2.
- [2] J. Jung, W. Lee, W. Kang, E. Shin, J. Ryu, and H. Choi, “Review of piezoelectric micromachined ultrasonic transducers and their applications,” *Journal of Micromechanics and Microengineering*, vol. 27, no. 11, 2017.
- [3] I. Zamora et al., “Miniaturized 0.13- μm CMOS Front-End Analog for AlN PMUT Matrices,” *Sensors*, vol. 20, no. 4, article 1205, 2020. doi: 10.3390/s20041205.
- [4] I. Zamora et al., “Monolithic Single PMUT-on-CMOS Ultrasound System with +17 dB SNR for Imaging Applications,” *IEEE Access*, vol. 8, pp. 142785–142794, 2020.
- [5] G. Matrone, A. Ramalli, and P. Tortoli, “Ultrasound B-Mode Imaging: Beamforming and Image Formation Techniques,” *Applied Sciences*, vol. 9, no. 12, 2019.
- [6] I. Zamora et al., “Phased Array Based on AlScN Piezoelectric Micromachined Ultrasound Transducers Monolithically Integrated on CMOS,” *IEEE Electron Device Letters*, vol. 43, no. 7, pp. 1113–1116, July 2022. doi: 10.1109/LED.2022.3175323.
- [7] SilTerra Malaysia Sdn. Bhd. *SilTerra official website*. Disponible en: <https://www.silterra.com>.
- [8] SilTerra Malaysia Sdn. Bhd. Documentación de tecnología CMOS HV de 0.13 μm y PDK asociado utilizado en el entorno de diseño.
- [9] E. Ledesma et al., “A Compact 0.5 mm² Immersed Nondestructive Ultrasound System Based on a PMUT-on-CMOS Array,” *IEEE Sensors Journal*, vol. 25, no. 24, pp. 43931–43940.
- [10] E. Ledesma et al., “A Low-Power PMUT-on-CMOS System for Future Medical Ultrasound Applications,” in *2025 IEEE International Ultrasonics Symposium (IUS)*, Utrecht, Netherlands, 2025, pp. 1–3. doi: 10.1109/IUS62464.2025.11201863.
- [11] Cadence Design Systems. *Virtuoso Design Environment*. Herramienta utilizada para el diseño esquemático y físico.
- [12] Siemens EDA. *Calibre Verification Platform*. Herramienta utilizada para verificación DRC, LVS y extracción.

- [13] A. Hastings, *The Art of Analog Layout*. 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2006.