

Trabajo Fin de Grado
Grado Electrónica, Robótica y Mecatrónica

Diseño de “buffers” para el control de píxeles en
sensores ToF

Autor: Antonio Jesús Balmaseda Delgado

Tutor: Ángel Rodríguez Vázquez

Dep. de Electrónica y Electromagnetismo
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2017



Proyecto Fin de Carrera
Ingeniería de Electrónica, Robótica y Mecatrónica

Diseño de “buffers” para el control de píxeles en sensores ToF

Autor:

Antonio Jesús Balmaseda Delgado

Tutor:

Ángel Rodríguez Vázquez

Catedrático de Universidad

Dep. de Electrónica y Electromagnetismo

Área de Electrónica

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2017

II

Proyecto Fin de Carrera: Diseño de “buffers” para el control de píxeles en sensores ToF

Autor: Antonio Jesús Balmaseda Delgado

Tutor: Ángel Rodríguez Vázquez

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2017

El Secretario del Tribunal

A mi familia

A mis maestros

Agradecimientos

A mi familia y amigos por su apoyo en los momentos más duros.

Al personal de Anafocus por dejarme usar sus equipos y su incomparable ayuda.

A Ana por volcarse tanto conmigo y ayudarme mucho más allá de lo que cualquier otro hubiera hecho.

Y en especial, por supuesto a Irlanda, pues sin ese trocito de ella, esto hubiera sido simplemente imposible.

Antonio Jesús Balmaseda Delgado

Sevilla, 09, 2017

La tecnología de tiempo de vuelo se muestra hoy como una muy buena opción para 3D 'imaging'. Sin embargo, los sensores con esta tecnología deben disponer de todos los componentes que aseguren un buen funcionamiento. Actualmente varios sensores de este tipo se están desarrollando para realizar tareas de reconstrucción 3D en tiempo real.

Actualmente solo podemos captar la realidad mediante tecnología analógica, siempre que esto sea así, la interfaz analógico digital, siempre será no solo necesaria, sino indispensable. La necesidad de captar el mundo no es la única tarea a la que nos debemos enfrentar, mientras los sensores sean analógicos, nuestra comunicación con ellos también deberá ser de la misma naturaleza. Es por ello que el bufero y conversión de las órdenes digitales a analógicas será un importante punto a tener en cuenta en todas las cámaras digitales.

En este trabajo se hará un repaso global sobre las tecnologías usadas en sensores de tiempo de vuelo para posteriormente realizar el diseño de los buffers encargados de llevar las señales digitales a la parte analógica del sensor. El documento se desarrollará de esta manera:

- Revisión de tecnología y documentación.
- Modelado de la carga debida a la línea de pixeles del buffer.
- Diseño del buffer encargado de transmitir las señales de control al pixel.
- Mejoras en velocidad de todo el bloque de Buffer.

The field of 3D imaging is now the new step for imaging. Since sensors are barely the same, specifications are harder day by day to achieve a correct image with depth perception, different solutions have been given to develop a correct 3D image. Time specification has been a problem for time of flight solutions, since calculating distance based on speed of light is crucial.

Distortion problems still the same, electronics and their configuration affect the way the image is captured, a bad configuration can affect the image in several ways. The capture method can affect the result too; the way the shutter is shot can produce several distortions.

| | |
|--|-------------|
| Agradecimientos | VI |
| Resumen | VIII |
| Abstract | X |
| Índice | XI |
| Índice de Tablas | XIII |
| Índice de Figuras | XIV |
| Notación | XVII |
| 1 Introducción | 1 |
| 2.1. Terminología de sensores | 1 |
| 2.2. Organización del trabajo | 2 |
| 2 Sensores de Imagen: tecnologías | 5 |
| 2.1 CCD | 6 |
| 2.2 Sensores CMOS. | 11 |
| 2.3 CCDvsCMOS. | 13 |
| 3 Píxeles activos | 16 |
| 3.1 Pixel 3T. | 17 |
| 3.2 Pixel 4T. | 17 |
| 3.3 Pixel 5T. | 18 |
| 3.4 Pixel 9T. | 19 |
| 3.5 Modos de shutter: "Rolling shutter" y "Global shutter". | 19 |
| 3.5.1 Pros y contras. | 21 |
| 4 Time of Flight (ToF) | 24 |
| 4.1 Principio de funcionamiento. | 24 |
| 4.2 Aplicaciones. | 29 |
| 4.3 Sensores en ToF: Elección de pixel. | 29 |
| 5 Diseño de búferes de señales de control de un pixel para ToF: Cálculos teóricos | 30 |
| 5.1 Bloque de Control de fila. | 30 |
| 5.1.1 Pixel array: Layout del pixel y extracción. | 31 |
| 5.2 Control de fila analógico. | 33 |
| 5.2.1 Buffers | 34 |
| 5.2.2 Level shifters | 45 |
| 5.2.3 Biasing | 48 |
| 6 Diseño de búferes de señales de control de un pixel para ToF: Implementación | 50 |
| 6.1 Buffers | 50 |
| 6.1.1 Buffer sin control de slew | 50 |
| 6.1.2 Buffer con control de slew | 54 |
| 6.2 Level shifter | 56 |
| 6.2.1 Level shifter con niveles de salida programable | 57 |
| 6.3 Biasing | 60 |

| | | |
|----------|-------------------------------|-----------|
| 6.4 | <i>Implementación física.</i> | 64 |
| 6.4.1 | Level shifter | 66 |
| 6.4.2 | Buffers | 66 |
| 6.4.3 | Biasing | 67 |
| 7 | Anexo. | 70 |
| | Bibliografía | 77 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1. Ventajas e inconvenientes del modo Global Shutter | 22 |
| Tabla 2. Ventajas e inconvenientes del modo Rolling Shutter | 23 |
| Tabla 3. Resultados de tiempos de subida a y bajada de todas las señales. | 64 |

| | |
|---|----|
| 1. Notas originales de Boyle y Smith brainstorm meeting el 9 de septiembre de 1969, primer boceto de un CCD. [3] | 6 |
| 2. Unión p-n de canal enterrado. [4] | 7 |
| 3. Foto detector Simple. La luz entra a través de la puerta del dispositivo. [5] | 7 |
| 4. Efecto de la temperatura en la corriente oscura $e/\text{pix}/\text{sec}$. El parámetro presentado es pA/cm^2 a 293K. [4] | 8 |
| 5. Transferencia de carga de un sensor CCD en tres fases. [5] | 9 |
| 6. Creación de pares electrones hueco por impacto. [7] | 10 |
| 7. Comparativa photodiode vs Photogate. [8] | 11 |
| 8. Modelo de pixel pasivo (PPS). [9] | 11 |
| 9. Modelo pixel array pasivo (PPS) [10] | 12 |
| 10. Modelo de pixel array activo (APS). [10] | 12 |
| 11. Comparativa Complejidad 'fill factor'. [8] | 13 |
| 12. Tabla de comparación CCDvsCMOS. [9] | 15 |
| 13. Posición relativa estándar del control y el timing en un sensor. [11] | 16 |
| 14. Ilustración pixel 3T. [12] | 17 |
| 15. Ilustración pixel 4T. [11] | 18 |
| 16. Representación pixel 9T | 19 |
| 17. Cronograma de funcionamiento 'Rolling shutter'. [13] | 20 |
| 18. Cronograma 'Global Shutter'. [13] | 20 |
| 19. Ilustración de la captura en modo 'rolling shutter'. [13] | 21 |
| 20. Distorsión creada por capturar objetos muy veloces con el modo 'Rolling Shutter'. [13] | 22 |
| 21. Esquema funcional de un sistema ToF. [15] | 24 |
| 22. Sensor de profundidad por Fizeau, 1849. [17] | 25 |
| 23. Cronograma de un sistema con 'pulsed vision'. [15] | 25 |
| 24. Esquema de funcionamiento de un sistema ToF con 'Continuous Wave Modulation'. [14] | 26 |
| 25. Imagen capturada con una cámara ToF. [14] | 27 |
| 26. Imagen sin lectura de amplitud (izquierda), y captura de amplitud (derecha). [14] | 27 |
| 27. Imagen resultado de varias capturas a largo de un objeto en movimiento. [14] | 28 |
| 28. Pixel ToF | 28 |
| 29. Esquema del bloque de control de fila. | 31 |
| 30. Layout del pixel elegido. | 32 |
| 31. Bloque de control de fila analógico. Vista de cada slice. | 33 |
| 32. Buffer simple sin control de slew. | 34 |
| 33. Buffer con control de slew. | 35 |
| 34. Buffer con control de slew y preetapa. | 35 |

XIV

| | |
|---|----|
| 35. Representación de una pista. W ancho, L largo y t espesor. | 36 |
| 36. Modelo de capacidades parásitas de línea. | 37 |
| 37. Modelo de capacidades parásitas del mosfet. [20] | 37 |
| 38. Modelo pi de la línea. | 38 |
| 39. Representación de señal de control. | 38 |
| 40. Etapa de salida del buffer sin control de slew. | 41 |
| 41. Modelo etapa de salida + modelo pi. | 42 |
| 42. Etapa de salida del buffer con control de slew. | 44 |
| 43. Modelo de la etapa de salida del buffer con control de slew. | 44 |
| 44. Rampa de bajada de nivel alto a nivel bajo en la línea de transmisión. | 45 |
| 45. Level shifter con nivel alto programable y tierra común. | 46 |
| 46. Level shifter con nivel alto y bajo programables. | 46 |
| 47. Level shifter con nivel alto y bajo programable. | 48 |
| 48. Esquemático para el control de slew de subida. | 48 |
| 49. Esquema del ensayo DC del inversor con carga capacitiva. | 51 |
| 50. Prueba de simetría del inversor. | 52 |
| 51. Comparación entre transiciones, punto de corte corregido. | 52 |
| 52. Esquema del buffer con pre-etapa. | 53 |
| 53. Comprobación de tiempos del buffer. | 53 |
| 54. Interruptor CMOS. | 54 |
| 55. Esquema de buffer con control de slew. | 55 |
| 56. Esquema 'Level shifter' básico. | 56 |
| 57. Layout de la prueba DC del primer level shifter. | 57 |
| 58. Pruebas dc_OP Level shifter niveles programables. | 58 |
| 59. Layout Level Shifter con tierras separadas. | 58 |
| 60. Prueba transitoria level shifter más buffer | 59 |
| 61. Pruebas transitorias level shifter+buffer | 59 |
| 62. Tiempo de subida pruebas buffer + level shifter. | 60 |
| 63. Esquema de Biasing para el control de la pendiente de subida. | 61 |
| 64. Esquema de biasing para control de la pendiente de bajada. | 62 |
| 65. Simulación de todos los bloques con modelado de salida. | 63 |
| 66. Simulación de todas las pendientes para todas las señales usando todos los bloques. | 63 |
| 67. Layout del Biasing encargado del control de subida. | 68 |
| 68. Layout del biasing encargado de la pendiente de bajada. | 69 |

| | |
|--|----|
| Anexo 1. Salida Buffer transitoria a máxima velocidad. | 70 |
| Anexo 2 . Salida transotira del buffer a mínima velocidad. | 71 |
| Anexo 3. Modelo pi parametrizable. | 71 |
| Anexo 4. Test DC biasing | 72 |
| Anexo 5. Tiempo de subida mínimo del buffer. | 73 |
| Anexo 6. Test DC inversor CMOS. | 74 |
| Anexo 7. Test DC level shifter más buffer. | 75 |
| Anexo 8. Esquemático de todas las pruebas de tiempo de subida. | 76 |

| | |
|-------|---|
| TOF | Time of Flight |
| CMOS | Complementary Metal-Oxide Semiconductor |
| CCD | Charged Couple Device |
| MOS | Metal oxide semiconductor |
| SNR | Signal-to-noise ratio |
| EMCCD | Electro Multiplying CCD |
| CPU | Central Processing Unit |
| PG | Photogates |
| PPS | Passive Pixels |
| APS | Active Pixels |
| FPN | Fixed Pattern Noise |
| CDS | Correlated Double Sampling |
| RAM | Random Access memory |
| PPD | Pixel photodiode |
| ICP | Iterative closest Point |

1 INTRODUCCIÓN

Only priests and fools are fearless and I've never been on the best of terms with God.

- Patrick Rothfuss -

En las aplicaciones de tiempo de vuelo se trabaja con el dato conocido de la velocidad de la luz y con su desplazamiento en fase, por ello se habla de capturar desfases del orden del picosegundo. El objetivo de los sensores utilizados para estas aplicaciones es percibir ese desfase, por tanto, capturar ráfagas rápidas de luz es esencial. Para hacer posible esto las arquitecturas más adecuadas son las de sensores de imágenes CMOS que garantizan unas velocidades de captura muy altas que se acercan a los requisitos de estas aplicaciones. Sin embargo, incluso en esta tecnología las arquitecturas convencionales empiezan a suponer más parte del problema que de la solución, resultando en un limitante físico en la velocidad a la que podemos capturar la luz. Esto se debe a que para tener una rápida captura de imágenes necesitaremos unas señales que controlen la captura que sean tanto o más rápidas que las velocidades a las que pretendemos calcular, de otro modo estaremos perdiendo información.

En cualquier caso, se demostrará que el tamaño del sensor influirá en la velocidad máxima de captura resultando en que mientras mayor “resolución” pretendamos alcanzar, menor deberá ser el área del sensor. Sin embargo, la minimización del área influirá en la cantidad de luz que somos capaces de captar y por tanto, en la resolución máxima de nuestro dispositivo. Sin entrar en detalles y anticipando al temario al lector se dirá que esto se debe a que las señales de control deberán viajar por el sensor para poder dar la orden de captura, descarga, etc.

A lo largo de este trabajo trataremos la dependencia de las velocidades máximas alcanzables en función de parámetros de diseño que se puedan configurar. No obstante, el lector ya será consciente de que sin aplicar ningún tipo de solución a los problemas mencionados habrá que alcanzar un compromiso entre el área del dispositivo y la velocidad máxima alcanzable.

2.1. Terminología de sensores

Signal to noise ratio (SNR): Es la relación entre la amplitud de una señal y el ruido que la ensucia.

Serial Register: Es una fila extra que se añade al fondo de la matriz en la tecnología CCD que se encarga de almacenar la información antes de liberarla a la lectura.

Photogates: Dispositivo que mide el tiempo entre eventos y que interrumpe un haz infrarrojo.

Fixed Pattern Noise: Patrón constante de ruido lateral no uniforme en un sistema con píxeles o múltiples detectores.

Correlated Double Sampling (CDS): CDS es una técnica que consta en tomar dos muestras de una señal muy cercanas en el tiempo y extraer la primera señal de la segunda, para eliminar el ruido correlado de baja frecuencia.

Fill Factor: Hace referencia a la relación entre el área sensora del pixel y el área total del mismo.

Eficiencia cuántica: Es la relación entre fotones que inciden en el sensor y los fotones que realmente se convierten en electrones.

Fotodiodos: Diodos que cambian su estado dependiendo de la luz incidente.

Blooming: Error en la captura de una imagen cuando la carga en un pixel excede el nivel de saturación y comienza a

Smearing: Es una señal no deseada que aparece de forma vertical en los sensores cuando la imagen ya generada es movida a los registros de almacenamiento.

Framerate: Velocidad del dispositivo haciendo capturas consecutivas.

Backscatter: Error que ocurre en las cámaras activas cuando la luz se refleja antes de llegar al objetivo.

Motion blur: Rastro dejado en la imagen por objetos en movimiento.

Corriente oscura: La corriente oscura se puede definir como una carga no deseable que se acumula en los píxeles de manera natural debido a causas como la temperatura

2.2. Organización del trabajo

En el capítulo 2 se presenta una introducción a sensores de imágenes junto con una comparación entre sensores CCD y sensores CMOS. En el capítulo 3 se presentan distintos tipos de píxeles con sus funciones explicadas, en este capítulo se elige una arquitectura de pixel con la que trabajar posteriormente, además en este capítulo se presentan los modos de funcionamiento. En el capítulo 4 exploraremos el funcionamiento básico de la tecnología Time of Flight y sus aplicaciones. En el capítulo 5 llegaremos a la simulación de varias arquitecturas que se pueden utilizar con el pixel anteriormente elegido para finalmente en el capítulo 6 llegar a su implementación y corroborar su funcionamiento. En el capítulo 7 exploramos nuevas formas de mejorar el sensor partiendo de la misma arquitectura base.

2 SENSORES DE IMAGEN: TECNOLOGÍAS

Fear tends to come from ignorance. Once I knew what the problem was, it was just a problem, nothing to fear.

- Patrick Rothfuss -

El sensor de imagen es el componente encargado de captar la información del entorno en forma de luz para poder construir una imagen posteriormente. Esto es posible gracias a la conversión de la luz en señales eléctricas que posteriormente pueden ser tratadas para la reconstrucción de la imagen del objetivo. La unidad básica del sensor de imagen es el píxel (del inglés ‘**picture element**’), que realiza la conversión de luz a señal eléctrica. Un grupo de píxeles capaces de detectar la radiación con cierta resolución espacial es lo que llamamos ‘imager’.

Con la ayuda de estos ‘imager’ se puede conseguir captar la escena objetivo ya sea en forma de imagen estática o de video en infinidad de aplicaciones, como fotografía, imagen 3D, imágenes para propósito médico, automovilístico o de seguridad, propósito industrial, lectores por reconocimiento de patrón, etc.

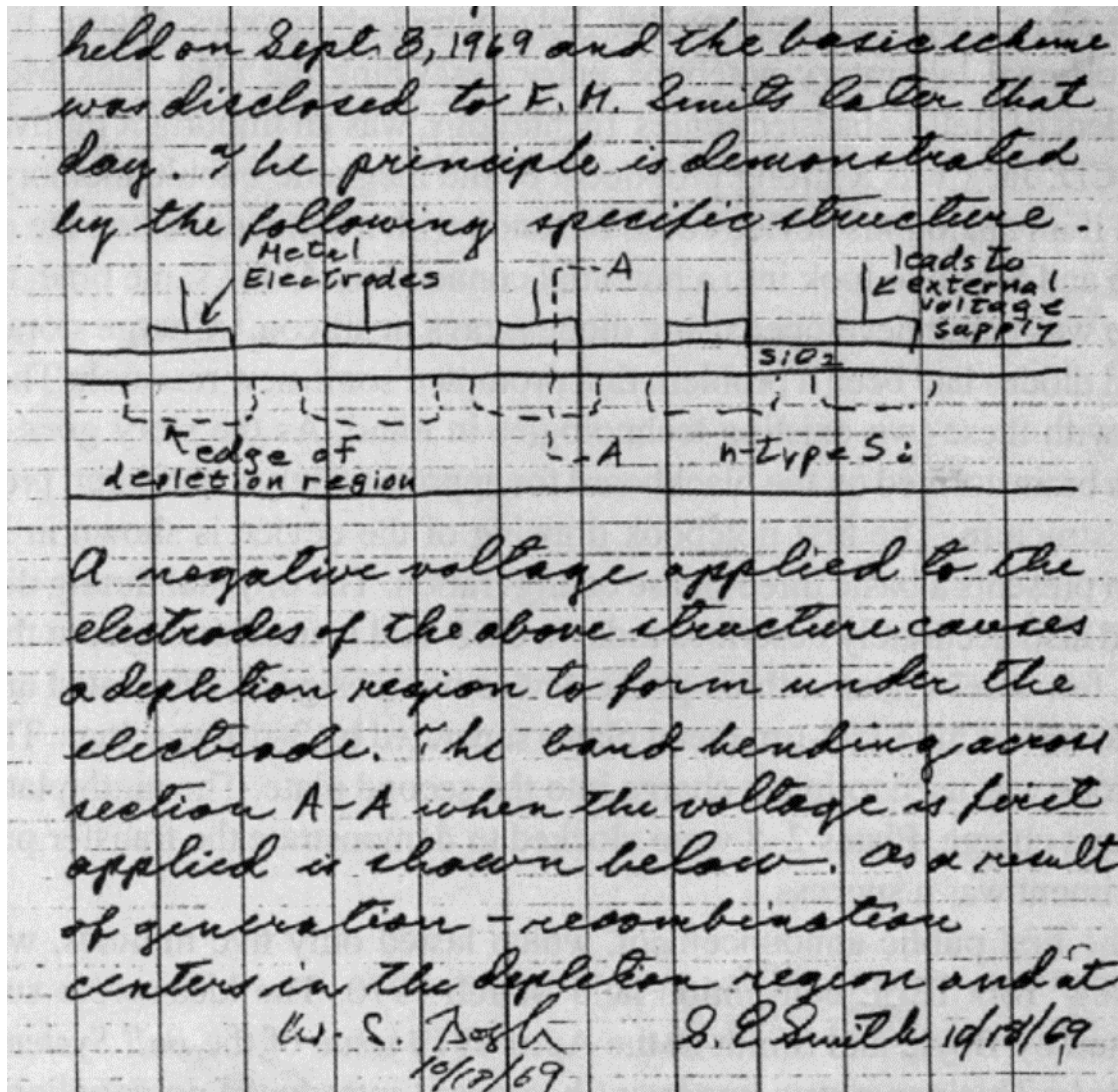
En lo referente a la tecnología para construir estos sensores de imagen, destacan más ampliamente los sensores **CCD** (Charged Couple Device) y **CMOS** (Complementary Metal Oxide Semiconductor), siendo la competencia entre ambas tecnologías algo bastante actual.

La primera vez que aparecieron los dispositivos **CMOS** fue alrededor de 1960 [1], sin embargo, estos dispositivos presentaban unas prestaciones pobres debidas a un alto ruido que deformaban la lectura de la señal, estos problemas fueron solucionados entre 1980 y 1990 [2].

Como es inevitable ahondar en una tecnología sin al menos nombrar la otra, a lo largo de este capítulo daremos una introducción de ambas. Como la primera tecnología en ser explotada y desarrollada fue la tecnología **CCD** empezaremos con ella, para posteriormente poder revisar con más perspectiva la trayectoria de la tecnología **CMOS**.

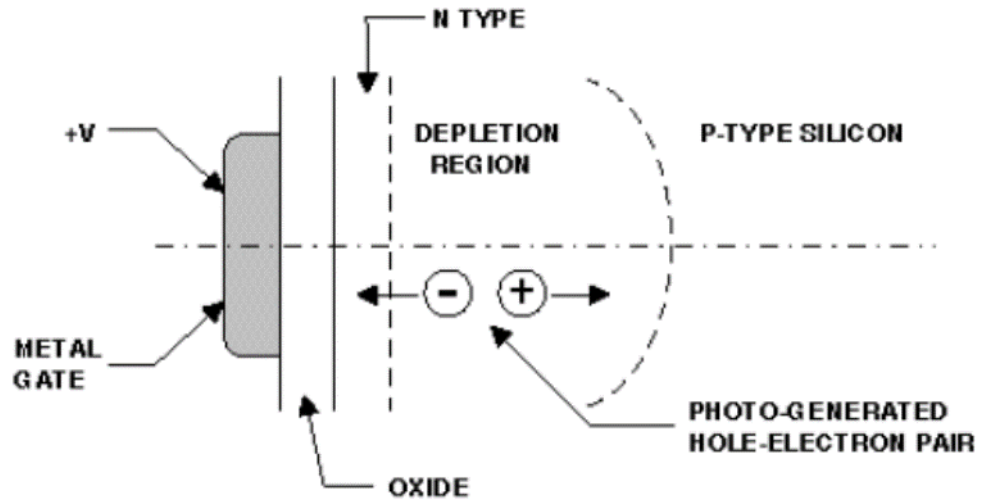
2.1 CCD

Inicialmente concebidos como registros de desplazamiento en 1971 por Goerge Smith y Willard Boyle, Michael Tompsett publica la primera patente para captura de imágenes [3].



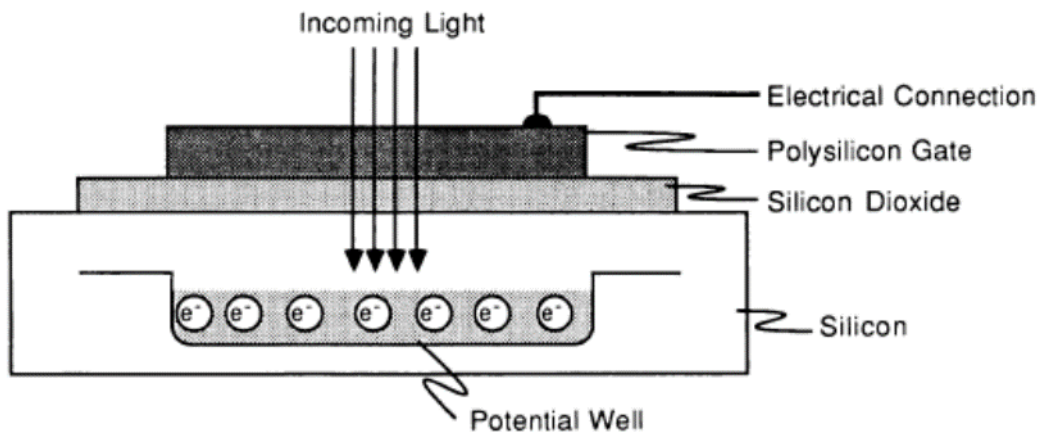
1. Notas originales de Boyle y Smith brainstorm meeting el 9 de septiembre de 1969, primer boceto de un CCD. [3]

Estos sensores están hechos por un dispositivo que mueve la carga eléctrica a la que se le añaden regiones fotoactivas, encargadas de convertir la radiación luminosa en carga eléctrica. Cada pixel está formado por una unión MOS p-n. Al polarizar en inversa esta unión, la región de depleción crece, creando un pozo de potencial en el que se almacenan los electrones [4].



2. Unión p-n de canal enterrado. [4]

La luz suele entrar en el dispositivo atravesando las puertas de las uniones p-n de modo que al incidir en la región fotosensible se crea un par electrón hueco. Típicamente la energía necesaria para crear este par ha sido de 1.1V. [5]



3. Foto detector Simple. La luz entra a través de la puerta del dispositivo. [5]

La luz con longitudes de onda menores a 400nm no pueden entrar al silicio y generar carga en el pozo. Se realizan mallas de estos dispositivos creando una matriz de píxeles que se encargan de almacenar la carga proporcionalmente a la imagen capturada. [5]

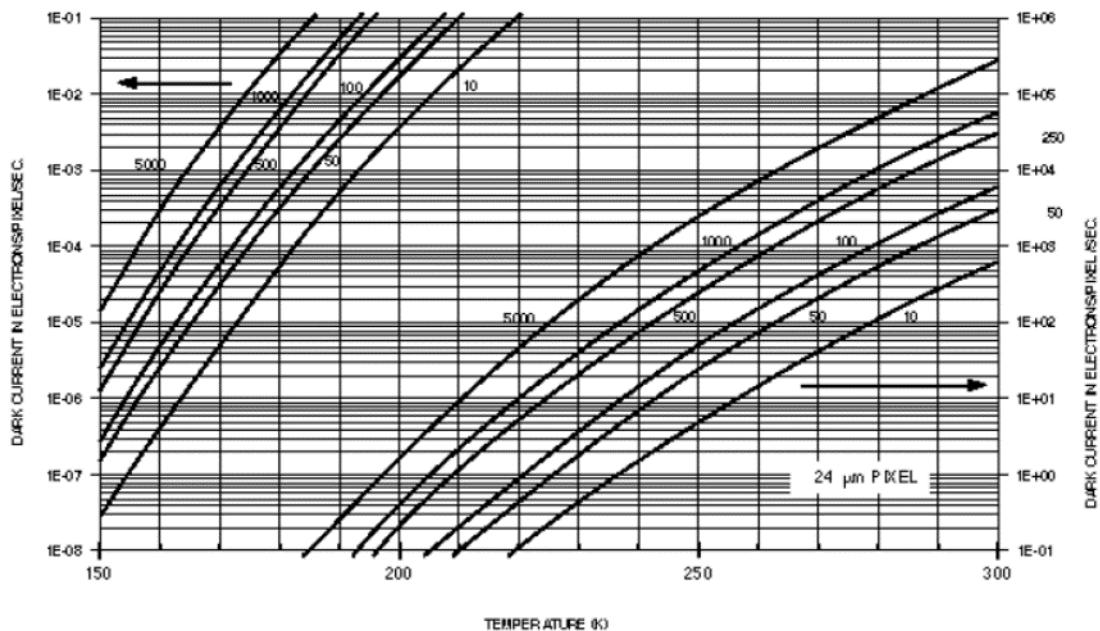
En resumen, usar esta disposición de metales nos reporta varios beneficios entre los que podemos contar [4]:

- La habilidad de crear un pozo de potencial en una región local.
- La habilidad de modular o controlar el potencial bajo la puerta.
- El almacenaje se localiza lejos de la unión del silicio con el óxido.

- Una baja corriente oscura nos permite almacenar la carga durante mucho tiempo.
- La carga puede ser generada ópticamente, inyectada eléctricamente o creada por partículas como rayos cósmicos, protones o fotones de alta energía.
- La capacidad de mover la carga de un electrodo al siguiente.

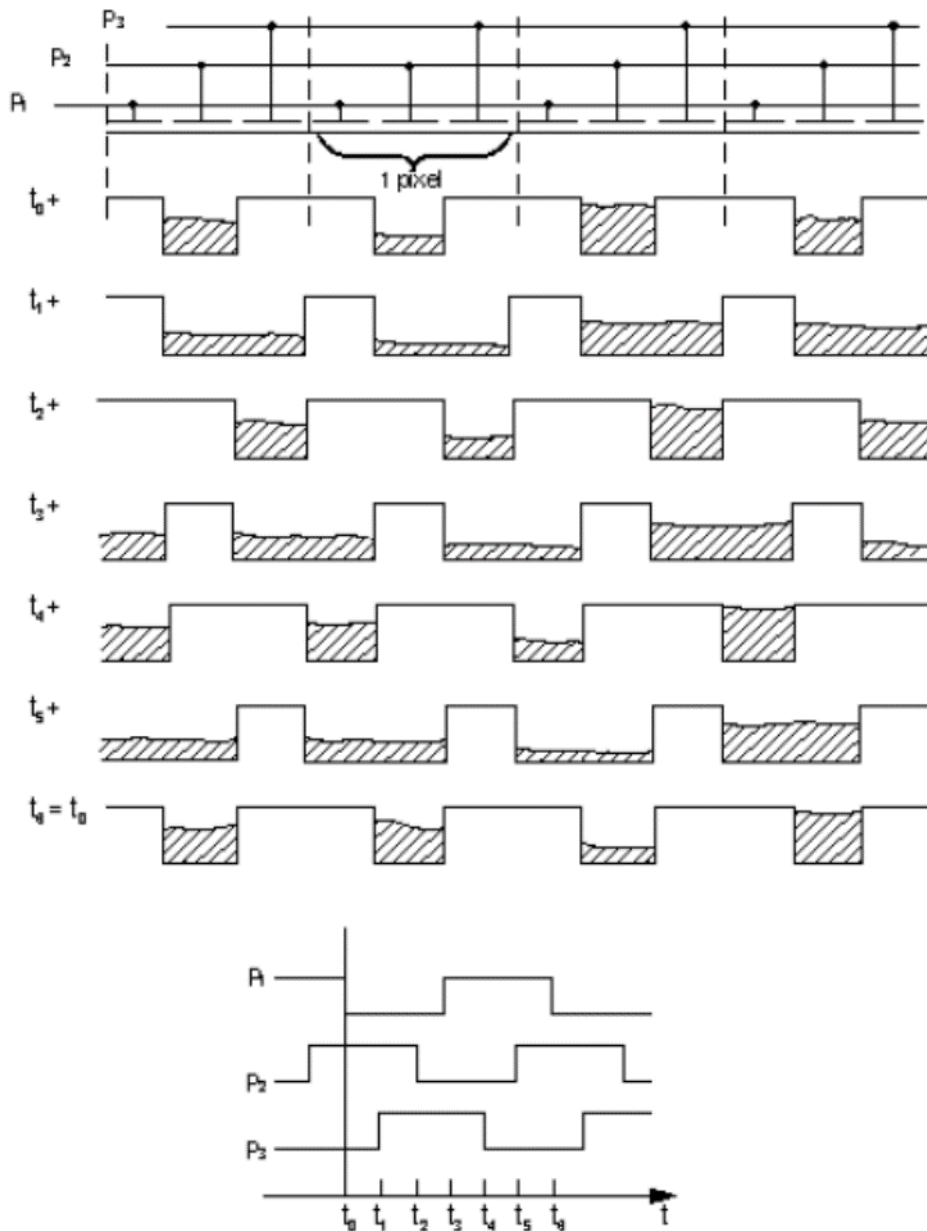
El punto referente a la corriente oscura es muy importante pues esta característica ha definido la forma en que los sensores CCD ha evolucionado y se ha especializado en aplicaciones como capturas de imágenes del espacio, en las que la energía de la radiación capturada es muy baja. Primero definiremos la corriente oscura.

Dado que la corriente oscura aparece siempre que la temperatura este por encima del cero absoluto, esta estará siempre presente. Si esta corriente es muy alta pronto enmudece la información capturada, e incluso puede llegar a saturar los pozos, de manera que toda información se pierde. Por suerte la corriente oscura depende fuertemente de la temperatura del dispositivo y del proceso, de manera que es posible mitigarla en gran medida simplemente enfriando el dispositivo hasta llegar a un 'signal-to-noise ratio' (SNR) aceptable. A continuación, se presenta una tabla que representa la corriente oscura en relación con la temperatura. [4]



4. Efecto de la temperatura en la corriente oscura e/pix/sec. El parámetro presentado es pA/cm² a 293K. [4]

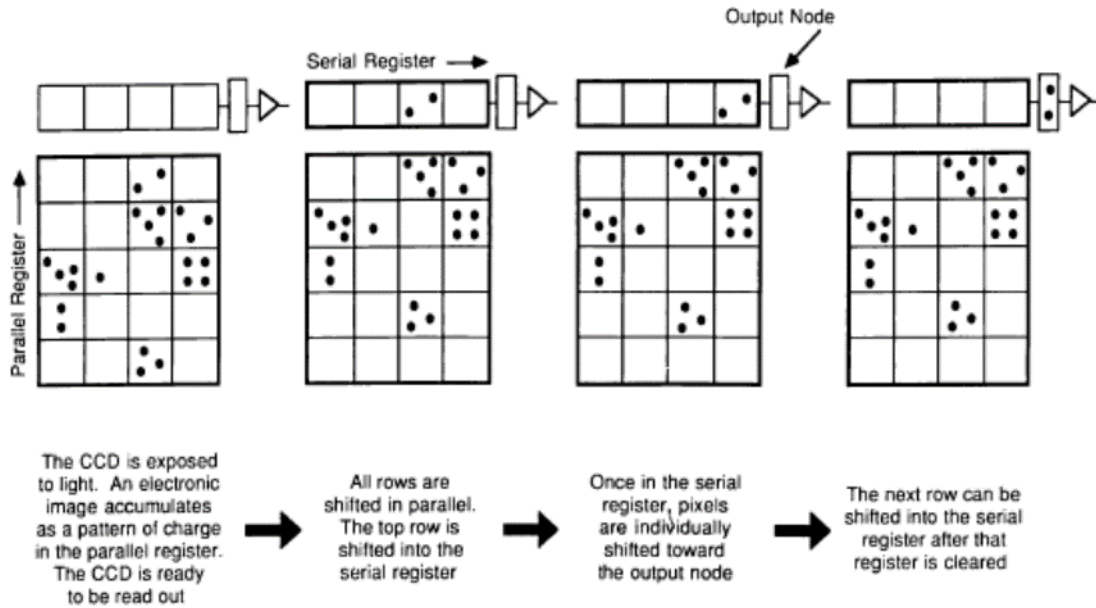
Para mover esta carga, se aprovecha el potencial del pozo, de manera que al haber una diferencia de potencial entre un pozo y el siguiente la carga se ve atraída hacia el pozo más profundo. De esta manera al coordinar la polarización de los distintos pozos, somos capaces de mover la carga, este proceso se conoce como 'shift'. En la figura anterior podemos ver un ejemplo ilustrativo de cómo se realiza el movimiento de la carga. [5]



5. Transferencia de carga de un sensor CCD en tres fases. [5]

Una vez la luz ha creado la carga y somos capaces de moverla a lo largo de la matriz, podemos proceder a la lectura de la misma para su procesado, se conoce a este proceso como proceso de lectura o ‘read-out’. La lectura se realiza primero por columnas y posteriormente por filas, de manera que las filas se van descargando en lo que se denomina ‘Serial Register’ [5] donde además, se puede dar la conversión de carga a tensión [6]. Aunque se podría ahondar mucho en el tema, en este documento solo se presentaran los datos más relevantes y básicos acerca del funcionamiento de los sensores CCD, ya que no es el tema principal de este trabajo.

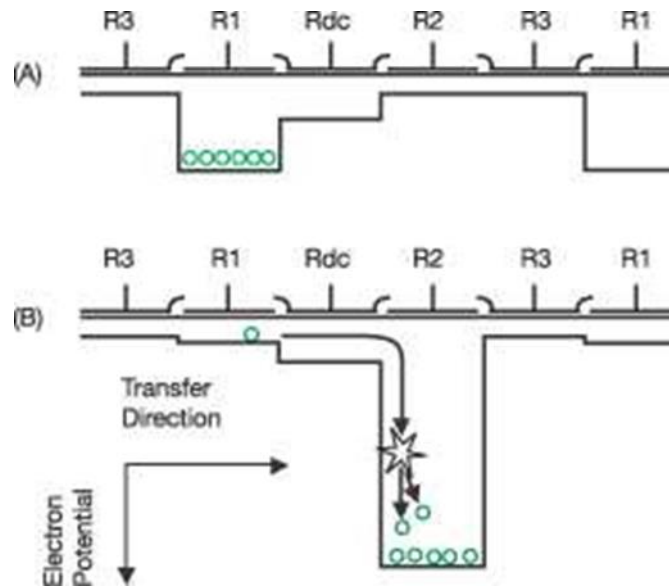
En la figura siguiente se puede comprender de manera intuitiva cómo funciona la lectura en este tipo de dispositivos englobando todo lo que se ha comentado hasta ahora.



2.6. Proceso de lectura de una matriz de pixeles en un sensor CCD. [5]

La amplificación se da al final de la lectura, esto tiene ciertas consecuencias, pero dependiendo de la aplicación que busquemos, puede ser interesante hacer la amplificación dentro de la matriz de pixeles. Se puede conseguir una amplificación dentro de la matriz con un sensor con ligeras modificaciones llamado ‘Electro Multiplying’ CCD (EMCCD). Este tipo de sensor ofrece una gran sensibilidad con ruidos de lectura menores a $10e^{-}$, a cambio de una lectura más lenta. Este tipo de sensores son especialmente eficaces para cámaras de larga exposición, donde la velocidad no es un limitante. Gracias a esta amplificación (o multiplicación) el ruido de lectura deja de ser limitante en la sensibilidad del sensor. [7]

Para conseguir esta multiplicación se recurre a un efecto que se basa en la creación de nuevos pares electrones-hueco por impacto. Como hemos visto anteriormente, el movimiento de la carga a lo largo de la matriz de pixeles, se realiza gracias a las diferencias de potencial entre los pixeles próximos entre sí. Si a la hora de mover la carga, creamos una gran diferencia de potencial entre dos pozos próximos se crean nuevos pares electrones huecos gracias a la fuerza del impacto de la carga. En la siguiente figura se puede observar una ilustración que muestra cómo se llevaría a cabo esta tarea a lo largo de una matriz de pixeles EMCCD. [7]



6. Creación de pares electrones hueco por impacto. [7]

2.2 Sensores CMOS.

Cuando hablamos de sensores CMOS podemos hablar de que se encuentran divididos en 4 bloques principales.

-‘Pixel array’.

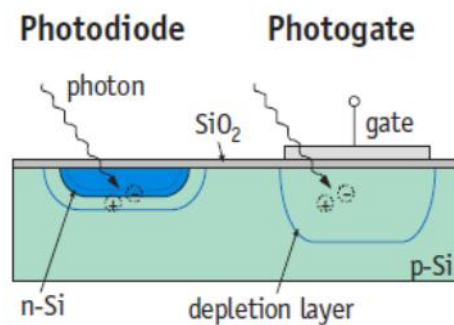
-‘Analog Signal processors’.

-‘Row&Column selector’.

-‘Timing and control’.

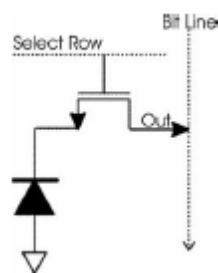
La función básica del sensor es la misma que en los sensores CCD, captar la radiación lumínica, con la diferencia de que en los sensores CMOS se hace la conversión directamente en diferencia de tensión.

A la hora de captar la luz, un fotodiodo no es la única opción, también pueden usarse fotogates (PG) que recolectan protones foto-generados pero solo cuando está siendo polarizado, de manera que permite tener el control sobre la captura de la luz. En la siguiente imagen podemos ver un corte transversal de estos dos captadores.



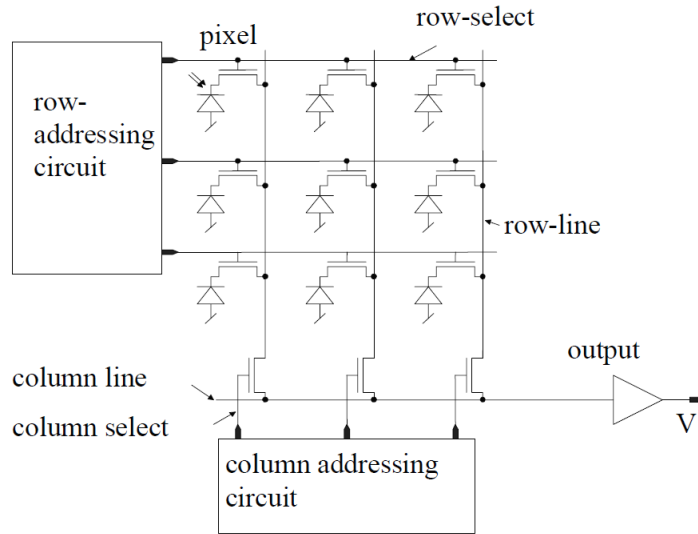
7. Comparativa photodiode vs Photogate. [8]

Al principio los primeros sensores CMOS se basaban en fotodiodos y usaban píxeles pasivos (PPS), es decir sin amplificación individual, y un transistor para conectar la línea de ‘readout’ [9] tal y como se muestra en la siguiente imagen.



8. Modelo de píxel pasivo (PPS). [9]

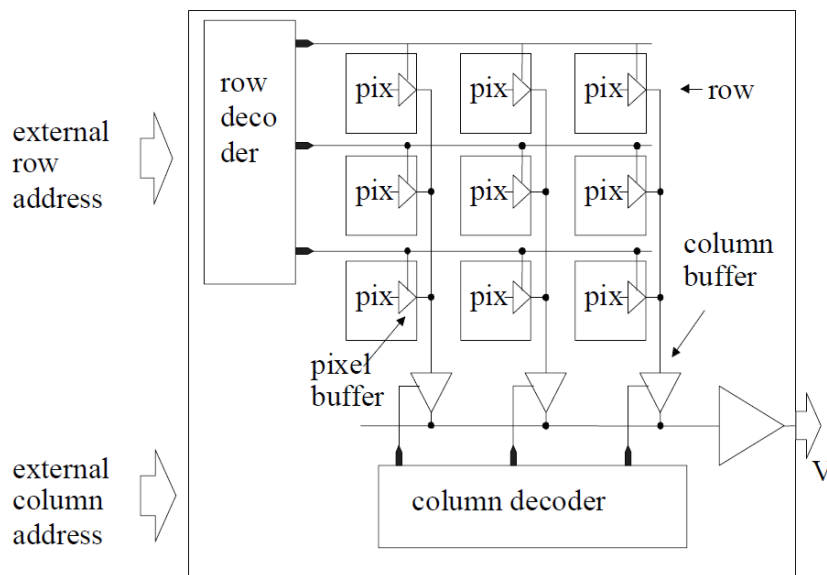
De modo que toda la matriz estaba compuesta por esta configuración, en la figura siguiente se muestra como se conformaría un sensor basado en píxeles pasivos.



9. Modelo pixel array pasivo (PPS) [10]

Sin embargo con el tiempo, los sensores CMOS comenzaron a usar un concepto muy interesante llamado pixel activo (APS), en el que la carga foto-generada se amplifica en el mismo pixel y la señal amplificada se transmite al ‘array’ del sensor. Una de las ventajas de este concepto es la supresión del ruido generado en la pista de la señal de lectura, en contraste con los pixeles activos los pixeles sin amplificación dese entonces se comenzaron a denominar pixeles pasivos.

Pese a esta amplificación individual, presentan una baja disipación pues solo se activan durante la lectura. Como contrapartida, estos pixeles presentan un gran ruido de tipo ‘fixed pattern noise’ (FPN), que se soluciona con el uso de ‘Correlated Double Sampling’ (CDS) que eliminan casi por completo las variaciones de disparo que provocan el offset [8]. En la siguiente figura podemos ver cómo sería la estructura de un sistema con pixel activo.



10. Modelo de pixel array activo (APS). [10]

Es necesario tener en cuenta el ‘fill factor’ y la eficiencia cuántica, pues al introducir electrónica encargada de la amplificación en cada pixel, estamos disminuyendo el área sensora por pixel [9], convirtiéndose así en un

índice de eficiencia a tener en cuenta.

2.3 CCDvsCMOS.

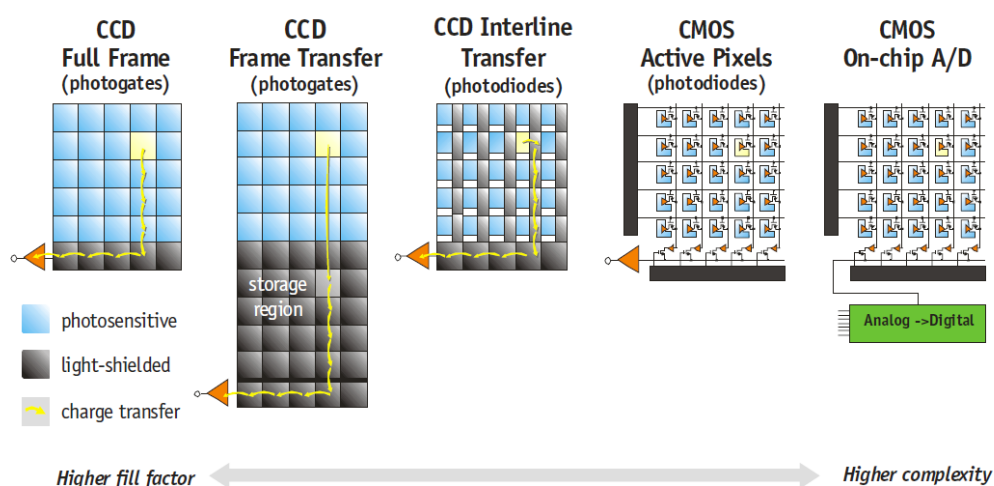
La primera vez que aparecen los CMOS fue en la década de los 60. Sin embargo, los sensores CMOS con los que se trabaja hoy día son fruto del trabajo a lo largo de la década de los 80. [1]

El problema de los primeros sensores CMOS era la cantidad de ruido que estropeaba las señales captadas, hasta el punto de reducir las aplicaciones donde podía ser útil. Estos errores se fueron corrigiendo entre los 80 y 90. [2]

Conociendo sus problemas iniciales nos preguntamos porque resultaba tan atractivo seguir investigando sobre ellos. El atractivo de los sensores CMOS reside en su bajo coste y en la integración de circuitería y componentes dentro del sensor, que hacen posible de nuevo la reducción de costes y de área total [6]. Estos sensores presentan un bajo consumo y suelen usarse en cámaras digitales y escáneres, gracias a lo cual han tenido un gran impacto en el siglo 21. Dada la demanda de teléfonos con cámaras de fotos integradas [6], su crecimiento se vio impulsado por grandes mercados como las memorias de estado sólido para CPUs, los circuitos integrados de propósito general y ahora los sensores de imagen.

Los sensores de imagen son los encargados de capturar la luz incidente, estos pueden estar formados por fotodiodos, que son semiconductores que reaccionan a las partículas de energía y fotones que son absorbidos para generar una tensión proporcional a la energía incidente [9].

Teniendo en mente los problemas de fill-factor y complejidad que se pueden ocasionar por la inclusión de amplificación en cada pixel es interesante ver una comparativa. En la figura (11) podemos apreciar una relación entre las arquitecturas utilizadas en los sensores CCD y CMOS teniendo en cuenta estos problemas.



11. Comparativa Complejidad 'fill factor'. [8]

No podemos perder de vista, que al usar estructuras más complejas podemos utilizar menor área para la misma operación y por tanto implementarlo todo en un solo chip.

Pero la estructura del pixel no es lo único que tiene un impacto en cómo se comportará finalmente el sensor. Otro de los aspectos que influye enormemente en el comportamiento del sensor es el modo de lectura. Hay muchos tipos de lectura, estos se usarán de manera diferente dependiendo de la aplicación para la que se vaya a utilizar el sensor [8]. Con el método de 'readout' se busca cumplir lo siguiente:

1. Baja disipación de energía.
2. Alta resolución.
3. Linearidad.
4. Biasing estable.
5. Bajo ruido.
6. 'High injection efficiency'
7. Tamaño pequeño de pixel.
8. Amplio rango dinámico.

Mientras que con las arquitecturas de pixel activos conseguimos todos los puntos excepto el tres y el siete, con las estructuras de pixel pasivo no podemos conseguir ni el tres, cuatro, seis, ni el ocho.

Con la información relatada hasta ahora se puede hacer una comparación entre ambas tecnologías y discernir cuáles son los puntos fuertes de cada una de ellas. Las diferencias básicas entre CMOS y CCD incluye funciones en chip, de tipo de salida y de sensor 'pixel array' [6].

Si nos centramos en la tecnología CCD, debemos tener en cuenta que un sensor CCD con una función básica estándar puede consumir varios wattios [8]. Si examinamos los pixeles, hemos visto que mientras el sensor CCD convierte los fotones en carga usando el efecto fotoeléctrico, en el sensor CMOS cada pixel convierte los electrones en una señal de tensión [6]. Teniendo en cuenta el funcionamiento, un sensor CCD operando con una función básica puede llegar a consumir varios wattios. [9]

Como contrapunto un sensor CCD posee un mayor 'fill factor' con tamaños de pixeles más pequeños, consiguiendo usar formatos más grandes que en los sensores CMOS. Por otro lado, un sensor CCD no puede ser integrado individualmente con la lectura analógica ni con la electrónica de control digital, tampoco es capaz de operar en ciertos rangos de iluminación y el ruido aumenta drásticamente con la temperatura. Además, al final de los años noventa, algunas tecnologías CCD incluyendo la transferencia de carga o las microlentes fueron no solo sustituidas sino optimizadas por tecnología CMOS. Otros de los problemas típicos de los sensores CCD son el 'Blooming' y el 'smearing'. [9]

Los sensores empezaron a ser una alternativa a partir de los años 90 [9]. Los sensores CCD siguen siendo superiores en calidad de imagen y relación señal ruido. En definitiva, las diferencias entre los sensores CMOS y CCD son:

La tecnología de fabricación del sensor CMOS ya existía, y estaba estandarizada mientras que la tecnología de fabricación de los sensores CCD es muy especializada.

La lectura de la tecnología CMOS es más flexible gracias a los esquemas filas-columna, que nos permite varios modos de lectura. En general la lectura filas-columna tiene la ventaja de menor consumo a alta velocidad.

La tecnología CMOS presenta un consumo menor ya que puede operar con una fuente de menor energía que la tecnología CCD. Además, el esquema fila-columna y la funcionalidad integrada ayuda a reducir la alimentación del sistema. Esto es especialmente importante para aplicaciones de lectura de alta velocidad.

Integrar los circuitos en un chip puede simplificar toda la electrónica de una cámara, la conversión analógica/digital y el procesamiento de la imagen puede además ser implementada, resultando en un sensor con una interfaz completamente digital.

Centrándonos en la tecnología CMOS, esta goza de dos modos de funcionamiento ‘global shutter’ y ‘rolling shutter’. Esto es, que se puede capturar y extraer la información de toda la matriz de píxeles a la vez (‘global shutter’) o bien fila a fila (‘rolling shutter’).

| CCD | CMOS |
|--------------------------------------|-----------------------------|
| Lower noise | Low power consumption |
| Smaller pixel size | Single power supply |
| Lower dark current | High Integration capability |
| 100% Fill Factor | Lower cost |
| Higher sensitivity | Single master clock |
| Electronic shutter without artifacts | Random access |

12. Tabla de comparación CCDvsCMOS. [9]

En el próximo capítulo se ahondará más en el tema de los modos de shutter, pues cada modo tiene sus ventajas y desventajas.

La tecnología CMOS nos da la oportunidad gracias a sus velocidades de trabajo de trabajar en captura de imágenes 3D haciendo usos distintos del flujo normal de captura de imagen. Uno de los tipos de recreación de imagen 3D es la tecnología basada en tiempo de vuelo (ToF) que trata de reconstruir la imagen gracias al valor conocido de la velocidad de la luz. En este trabajo nos centraremos en las aplicaciones de ToF.

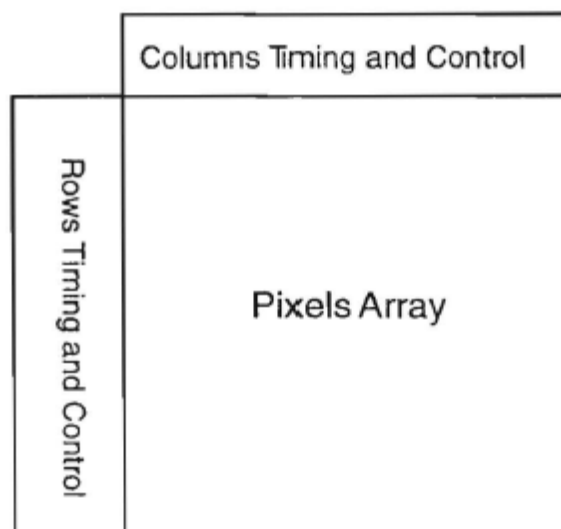
3 PÍXELES ACTIVOS

No, we're not playing God. We're only attempting to set things right.

- Clyde DeSouza -

El corazón de todo sensor es el ‘array’ de píxeles, en cada píxel hay una zona fotosensible y varios transistores encargados de bufear la señal captada hasta la línea de lectura. El tamaño de esta fila determinará la resolución de nuestro sensor.

Un píxel activo **CMOS** es aleatoriamente accesible de manera similar a una ‘random-access-memory’ (**RAM**). La accesibilidad del píxel está sujeta a los ‘encoders’, uno por fila y otro por columna, de este modo cada píxel posee su propia “dirección”. Estos dos ‘encoders’ representan el ‘timing’ y el control mínimo de un sensor en chip [11]. De este modo comprendemos que las señales de control y ‘timing’ a este nivel son de vital importancia para determinar la velocidad de captura de nuestra cámara. En la siguiente imagen se ilustra la disposición típica del control y el ‘timing’ con respecto a la matriz de píxeles.



13. Posición relativa estándar del control y el timing en un sensor. [11]

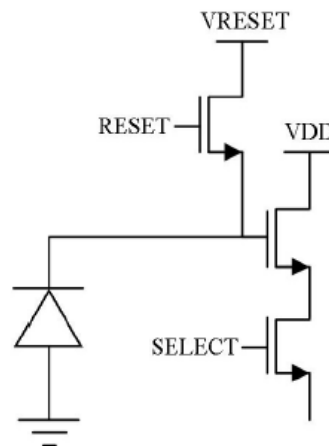
En este documento nos centraremos en estos dos aspectos, buscando la manera más eficiente de controlar los píxeles lograremos unas condiciones de ‘frame rate’ adecuadas para los modos que se van a tratar.

Pero antes de empezar con los detalles de esta búsqueda, a continuación, se presentan distintos tipos de píxeles, donde podremos apreciar donde atacan nuestras señales de control. Finalmente, tras esto, se profundizará en los modos de ‘shutter’.

3.1 Pixel 3T.

Antes de hablar de píxeles normalmente utilizados en sistemas de **ToF**, se explorarán esquemas de píxeles más tradicionales.

En la siguiente figura se muestra un pixel 3T (3 transistores) en el que se trata de captarla luz y enviarla por la rama OUT.



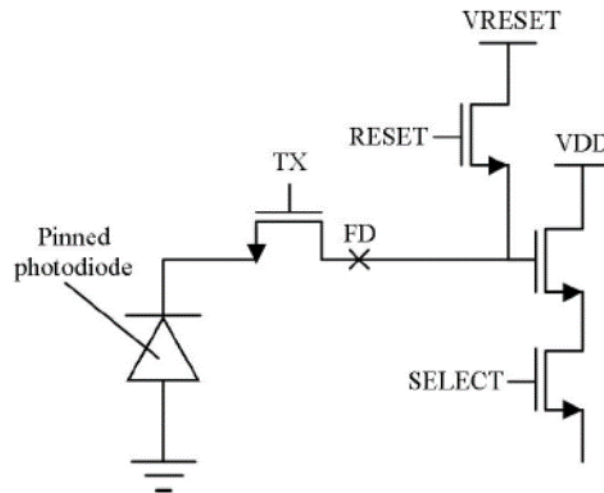
14. Ilustración pixel 3T. [12]

Este pixel es muy básico y consta tan solo de un foto-detector y tres transistores. El fotodiodo sirve, además, de integrador al almacenar la información del exterior en forma de carga. SF es un ‘source follower’ que seguirá la tensión proporcional a la carga almacenada en el fotodiodo. Estos son los dos únicos elementos encargados de modular la radiación lumínica en tensión que pueda ser interpretada por los dispositivos. El transistor Rst por su parte se encargará de limpiar la carga almacenada en el fotodiodo entre captura y captura, asegurándose de que no se contamine una captura con información de una captura anterior. Por último, el transistor Sel se encarga de dar paso a la tensión a la línea de salida, realizándose la lectura final cuando se activa.

Los amplificadores por pixel permiten que los píxeles 3T generen una buena señal de salida, sin embargo, el ruido que generan los relega por detrás de los sensores **CCD** [8]. No obstante, aún hoy día, se usan en diversas aplicaciones médicas que requieren que el sensor sea muy pequeño. Uno de los ejemplos son cámaras que pretenden utilizarse para explorar el cuerpo humano por dentro.

3.2 Pixel 4T.

En la siguiente ilustración podemos ver cómo sería el esquema de un pixel 4T.



15. Ilustración pixel 4T. [11]

En este caso se añade un transistor encargado de controlar cuando se integrará la carga almacenada en el fotodiodo. De esta forma tenemos un control directo sobre toda la matriz de píxeles. El funcionamiento del resto de transistores es análogo al funcionamiento en un pixel 3T. Al añadir más transistores el 'fill factor' baja, sin embargo, conseguimos a la vez mejorar los problemas de ruido [8].

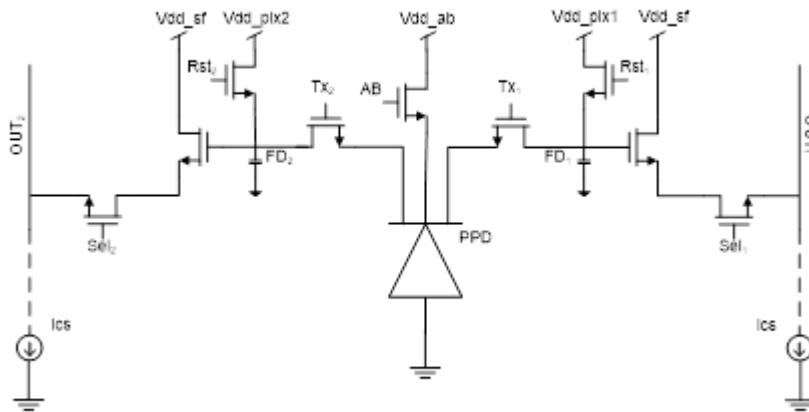
3.3 Pixel 5T.

En la siguiente imagen podemos ver el esquema de un pixel 5T.

Este pixel cuenta con el mismo 'trade-off' que el pixel 4T. La adición de un nuevo transistor viene por la necesidad de limpiar los posibles restos de carga en el fotodiodo, de esta manera conseguimos reducir aún más el ruido debido a residuos de antiguas capturas.

3.4 Pixel 9T.

Este pixel, añade al pixel 5T una nueva rama por donde almacenar una nueva captura. En la siguiente figura podemos ver una representación del pixel.



16. Representación pixel 9T

Como en los casos anteriores, añadir transistores reduce el ‘fill factor’, sin embargo, esta configuración nos permite sacar aún más partido del área utilizada. De nuevo, existe un transistor encargado de limpiar el fotodiodo (PPD) entre captura y captura, pero además la nueva rama utilizada nos permite permutar capturas entre las dos líneas.

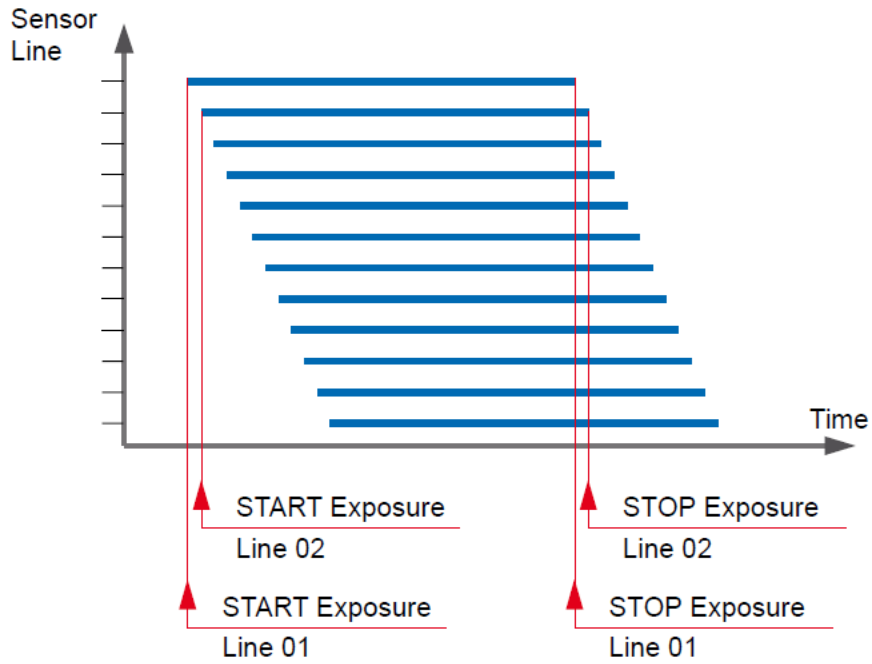
Esta configuración, aunque no es necesaria, es especialmente adecuada para aplicaciones **ToF**, pues nos facilita la comparación entre varias capturas, que resulta ser esencial para el cálculo de distancias y la reconstrucción 3D.

De nuevo el funcionamiento de los transistores es análogo a las configuraciones anteriores, con la salvedad de la permutación. De este modo mientras se está leyendo una de las ramas, podemos realizar otra captura con la rama opuesta.

3.5 Modos de shutter: “Rolling shutter” y “Global shutter”.

Ahora que conocemos las distintas arquitecturas de píxeles activos, debemos adentrarnos en los distintos modos de captura que podemos llevar a cabo con ellos. De manera general hay dos modos imperantes en los que podemos captar la imagen. ‘Rolling shutter’ y ‘Global shutter’.

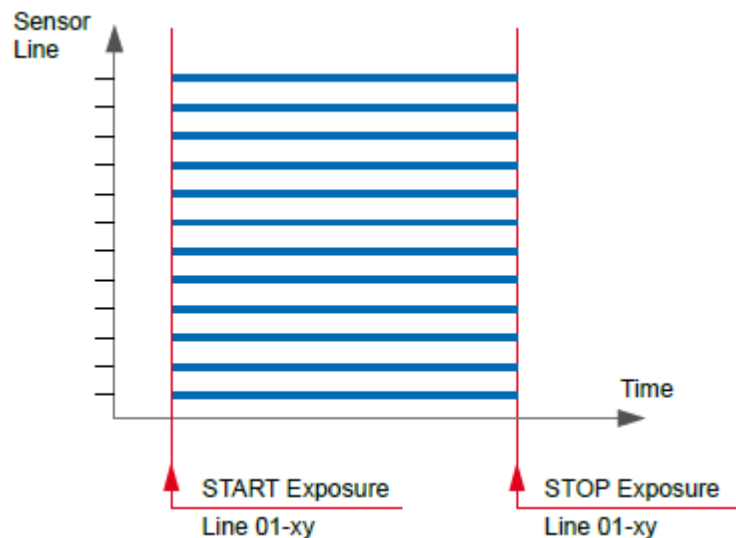
Cada uno de ellos tiene sus ventajas y desventajas. Empezando por el modo ‘Rolling shutter’, se basa en exponer una a una y de manera consecutiva las filas para captar la escena. En la siguiente imagen, podemos ver un cronograma bastante explicativo.



17. Cronograma de funcionamiento 'Rolling shutter'. [13]

Con este método, la exposición se va realizando de manera consecutiva, en este método pueden darse o no superposiciones de varias filas exponiéndose al mismo tiempo, dependiendo de la aplicación.

En cambio, en el modo 'global shutter', la exposición de las filas se hace de modo simultáneo, esto es, todas las filas se exponen a capturar luz en el mismo instante. En el siguiente cronograma podemos apreciar la diferencia entre ambos modos.



18. Cronograma 'Global Shutter'. [13]

Este modo de 'shutter' estuvo durante mucho tiempo vinculado únicamente a los sensores **CCD**, sin embargo, actualmente los sensores **CMOS** disfrutan de esta opción también [13].

Ahora que conocemos la diferencia entre usar un modo u otro, a continuación, exploraremos que ventajas y desventajas tiene un modo sobre el otro.

3.5.1 Pros y contras.

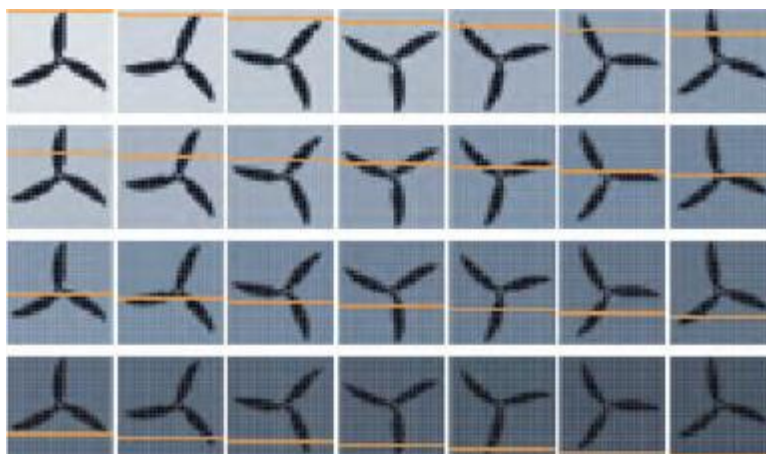
Si bien ambos modos pueden utilizarse en muchas aplicaciones, es inevitable que un modo prepondere sobre el otro en ciertas ocasiones. Vamos a presentar a continuación cuales son los problemas que pueden surgir usando los distintos modos en distintas aplicaciones.

El modo ‘rolling shutter’ goza de la ventaja de necesitar menos transistores para poder llevar a cabo su tarea, además al activarse los transistores de control consecutivamente, gozamos de un menor consumo que en modo ‘global shutter’. En el modo ‘global shutter’, sin embargo, observamos una demanda de corriente muy superior, al necesitarse activar muchos transistores al mismo tiempo derivando esto en que el tamaño de pixel mínimo para una aplicación que use ‘global shutter’ será alrededor de los 3,45 μm , mientras que en el caso de un pixel orientado a usar ‘Rolling Shutter’ está alrededor de los 0,9 μm . Por tanto el tamaño de los píxeles usados puede variar enormemente dependiendo del tipo de ‘shutter que se pretenda utilizar. Sobra mencionar además que si se pretende realizar una cámara capaz de operar en ambos modos debe tener por tanto al menos el tamaño necesario para utilizar el modo ‘global shutter’. Además el tamaño del pixel no es el único inconveniente cuando se pretende usar un modo de captura global, el inconveniente no solo radica en el tamaño del pixel, sino además en su complejidad, al necesitar más cantidad de transistores para operar, la complejidad del diseño se ve incrementada para poder mantener un control total del sensor.

Además de los problemas relacionados con el tamaño y complejidad del pixel para el modo de captura global, este modo presenta cierta desventaja durante las capturas de puntos con alta temperatura o de larga exposición, ocasionando la aparición de ‘hot pixels’, que no son más que píxeles que permanecen completamente blancos o negros sin, aparentemente reaccionar a la incidencia de la luz.

No obstante, el modo ‘rolling shutter’ presenta además sus desventajas, el inconveniente, que quizá es más llamativo aparece cuando se pretenden capturar objetos que realizan un movimiento a muy alta velocidad. Si las velocidades del movimiento son equiparables a la velocidad de exposición de las distintas filas, nos encontramos que cada fila está capturando distintas imágenes. En la ilustración siguiente podemos apreciar paso a paso esta problemática.

Siendo la franja amarilla la representación de la línea que se está exponiendo en cada momento.



19. Ilustración de la captura en modo ‘rolling shutter’. [13]

Al verse esta diferencia entre captura y captura, obtenemos al final una imagen distorsionada de la realidad. Con el modo ‘global shutter’, al exponerse todas las filas al mismo tiempo, la imagen capturada no se verá afectada por este efecto.

En las siguientes imágenes podemos observar un caso práctico de esta problemática y la distorsión que crea.



20. Distorsión creada por capturar objetos muy veloces con el modo 'Rolling Shutter'. [13]

Como se puede apreciar, la captura consecutiva de las filas ocasiona una distorsión en la imagen real. Aunque esta es la desventaja más llamativa, no es la única, a raíz del problema anterior nos percatamos que se produce un error de posición en la captura cuando el punto de captura se está moviendo, de modo que el punto de posición de la misma es incierto con estas capturas.

Por otro lado si hablamos de las ventajas de ambos modos, podemos concretar que gracias a la baja complejidad de los píxeles en el modo 'rolling shutter' podemos lograr una muy alta resolución en relación al formato utilizado, además al no solaparse muchas acciones en las filas al mismo tiempo, en este modo el ruido tiende a ser menor, pues la demanda de corriente será menor y se ensuciarán menos los canales durante la capturar. La posibilidad de usar píxeles pequeños en esta aplicación viene dada a que no es necesario usar una celda de memoria, pues cada exposición se hará justo antes de realizar de la lectura.

En cuanto al modo de 'global shutter' se presentan varias ventajas, además de la ya mencionada en que todas las filas se exponen al mismo tiempo, al usar píxeles más complejos en los que además tenemos una celda de memoria, no nos vemos atrapados al utilizar una aplicación concreta, por tanto este modo suele presentar más flexibilidad a la hora de usar distintas aplicaciones.

A continuación se presenta una tabla donde pueden verse las ventajas e inconvenientes de forma reducida.

'Global Shutter'

Tabla 1. Ventajas e inconvenientes del modo Global Shutter

| Ventajas | Desventajas |
|--|--|
| Todas las filas se exponen al mismo tiempo | Aparición de 'hot pixels' durante capturas con alta temperatura y larga exposición |
| Flexibilidad en aplicaciones | Requiere de una tecnología más compleja y de mayor tamaño del sensor |
| No se distorsionan los objetos en las capturas | Pixel mínimo relativamente grande $\sim 3,45 \mu m$ |

'Rolling Shutter'

Tabla 2. Ventajas e inconvenientes del modo Rolling Shutter

| Ventajas | Desventajas |
|--|---|
| Alta resolución con un formato pequeño | Los objetos dinámicos en las capturas se ven deformados |
| Menor ruido durante las capturas que en el modo ‘global shutter’ | El ‘shutter’ se mueve desde lo más bajo del sensor. |
| No hay necesidad de celda de memoria, estructura más simple permite tamaños pequeños de pixel $\sim 0,9 \mu m$ | Error posicional al capturar objetos en movimiento. |

4 TIME OF FLIGHT (ToF)

Any sufficiently advanced technology is indistinguishable from magic.

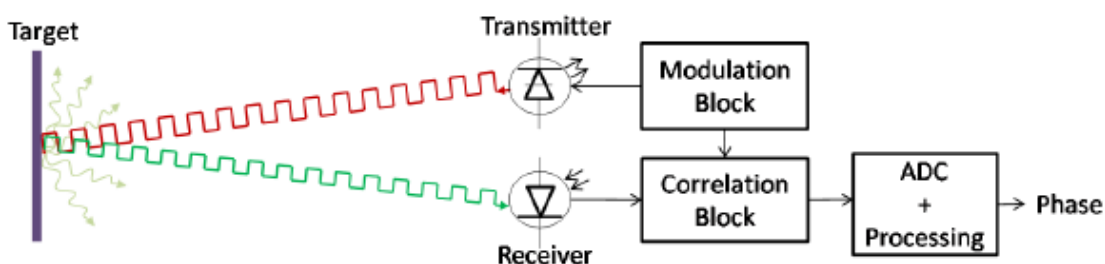
- Arthur C. Clarke-

En este apartado profundizaremos en las aplicaciones de tiempo de vuelo, donde se trata de hacer capturas de la escena en tres dimensiones, para ello deberemos hacer uso de píxeles activos del tipo que se han mencionado en apartados anteriores. Además de profundizar en el funcionamiento de esta tecnología, se mostrarán algunas de las aplicaciones en que puede ser útil. Al final de este apartado, tras un breve razonamiento se elegirá el pixel adecuado sobre el que se trabajará en el resto de este documento.

4.1 Principio de funcionamiento.

Como se ha comentado en apartados anteriores, los sistemas de visión se pueden clasificar en dos grandes grupos, activos o pasivos. Mientras que los sistemas pasivos se ocupan únicamente de capturar la luz de la escena, los sistemas activos también toman parte en la iluminación del objetivo gracias a un emisor, que normalmente proyecta luz indetectable al ojo humano.

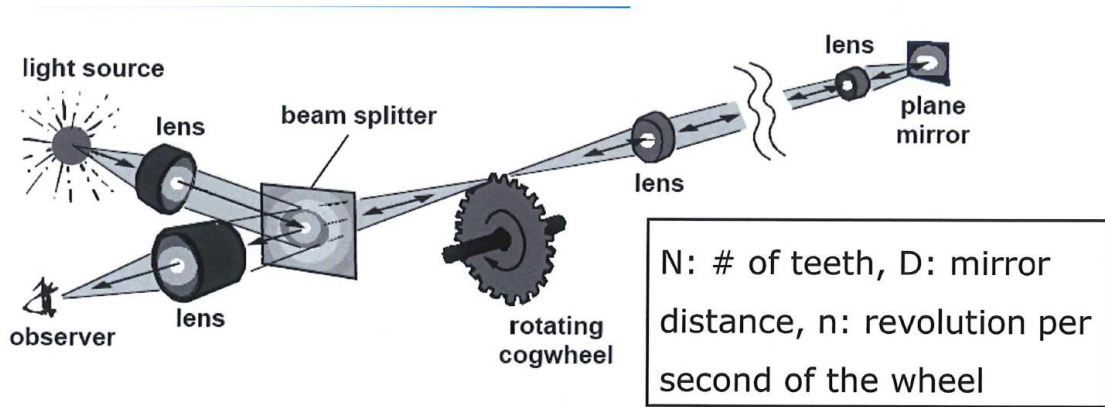
Un sistema de 'Time of Flight' (ToF) es un sistema que produce una imagen con profundidad, y cada pixel encierra en sí la información de la distancia punto a punto del objeto. Una cámara ToF se basa en iluminar la escena y, midiendo el desfase entre la luz proyectada y la reflejada adquirir el dato de la profundidad [14].



21. Esquema funcional de un sistema ToF. [15]

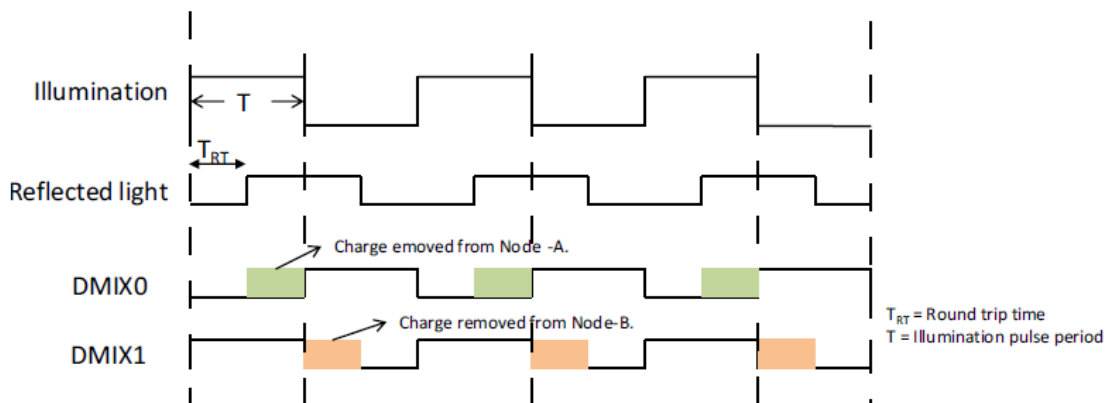
El funcionamiento de esta aplicación es posible gracias a la naturaleza activa del sistema. La luz emitida viaja

hacia el objeto y rebota parcialmente [16], al reflejarse la luz, se origina un desfase entre la luz emitida y la luz capturada posteriormente. Gracias a este desfase y al dato conocido de la velocidad de la luz, se puede conseguir calcular la distancia del objetivo hasta la cámara. Aunque esta aplicación pueda parecer relativamente nueva, el primer sensor de **ToF** fue publicado por Fizeau en 1849, a continuación, se ilustra un esquema. [17]



22. Sensor de profundidad por Fizeau, 1849. [17]

Si bien el sensor no parece tener relación con las cámaras actuales, el principio en el que se basa es el mismo. Como se ha comentado, el sistema se basa en emitir una luz para luego leer el desfase, de manera que el sensor determina “cuando” regresa la luz [16]. Hay varias formas de emitir la luz, un ejemplo es el ‘pulsed vision’ que se basa en iluminar la escena con un tren de pulsos lumínicos [17], normalmente con un láser de luz infrarroja. En la siguiente imagen podemos observar cómo sería el cronograma de un sistema con ‘pulsed vision’ usando un pixel 9T.



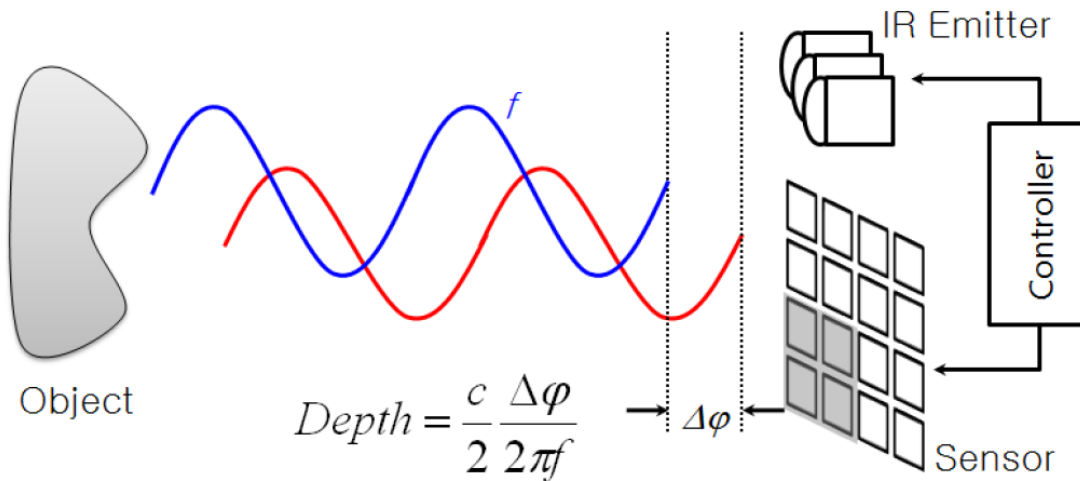
23. Cronograma de un sistema con ‘pulsed vision’. [15]

Usar un sistema activo nos proporciona ciertas ventajas, como que el sensor sea inmune a la luz ambiente y a las distintas propiedades ópticas de los objetos [16]. Además, no nos es necesario utilizar un “contraste” con el que comparar para obtener una imagen que muestre la realidad de la escena [18].

Esta tecnología posee numerosas aplicaciones como visión artificial, interacción con la escena, numerosas aplicaciones médicas, etc [19]. Utilizar la tecnología **ToF** para la captura de imágenes con profundidad trae numerosas ventajas consigo aparte de las mencionadas, por ejemplo [19]:

- Solo se requiere una cámara específica.
- No posee una computación manual de profundidad.
- La adquisición de la escena en 3D es a tiempo real.
- Hay poca dependencia con la iluminación de la escena.
- Casi independiente de la textura de la superficie de los objetos.

Ahora que conocemos las ventajas de usar esta tecnología es importante mencionar que ‘pulsed vision’ no es la una forma de iluminar la escena, otro de los ejemplos es la ‘Continuous Wave Modulation’ [19].



24. Esquema de funcionamiento de un sistema ToF con ‘Continuous Wave Modulation’. [14]

La elección de una sobre la otra trae consigo algunas ventajas y desventajas.

Mientras usamos ‘pulsed’ visión tendremos las siguientes ventajas y desventajas [19]:

- Medida directa de la escena
- Los pulsos de gran energía imitan la dependencia a la iluminación de la escena
- Iluminación y observación colineales.
- Se requiere una medida de tiempo de gran resolución
- Dificultad para genera pulsos con tiempos de subida y bajada suficientemente rápidos.

En cambio, usando ‘Continuous Wave Modulation’ [19]:

- Se pueden aplicar diferentes técnicas de modulación
- Rango y amplitud en imágenes simultaneas
- Varios tipos de generadores disponibles
- En la práctica se necesita reducir el ruido
- ‘Frame rate’ excede el tiempo de integración
- ‘Motion blur’ causada por el largo periodo de integración

A lo largo de este trabajo nos vamos a centrar en los sistemas **ToF** con ‘pulsed vision’. En la siguiente ilustración podemos ver una imagen capturada con esta tecnología.



25. Imagen capturada con una cámara ToF. [14]

Dependiendo de la escena que se pretenda capturar tendremos unas limitaciones u otras. La resolución de profundidad y el ‘frame rate’ vendrán limitados por el tiempo de integración de la cámara, por tanto, conviene bajar el ‘frame rate’ para objetos quietos. Sin embargo, para objetos móviles el tiempo de integración estará limitado. [14]

Además, este tipo de tecnologías necesitan un apoyo extra en cuanto a colores planos, pues si solo medimos el desfase antes mencionado al capturar un objeto con distintos colores, la cámara los obviará. Es por esto que además de capturar la distancia punto a punto también se realiza una medida de la amplitud de la luz reflejada. De este modo al capturar la luz infrarroja reflejada seremos capaces de observar estos distintos colores. En la imagen siguiente podemos observar debidamente esta problemática al intentar capturar la imagen de un tablero de ajedrez. [14]



26. Imagen sin lectura de amplitud (izquierda), y captura de amplitud (derecha). [14]

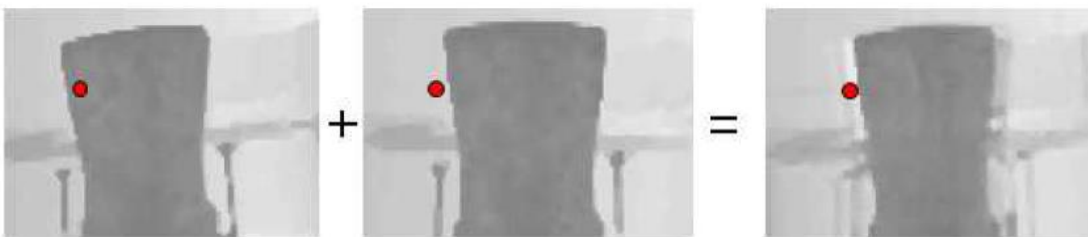
En la imagen de la izquierda podemos observar cómo se haría la captura de profundidad, esta imagen debe ser complementada con las distintas amplitudes de luz infrarroja para poder observar los detalles de un objeto que se encuentre con una cara plana a la misma profundidad.

Otro de los errores que conviene tener en cuenta es el ‘backscatter’, es un problema que aparece cuando parte de la luz emitida vuelve al sensor antes de que alcance al objetivo, originándose un objeto sólido donde no existe realmente.

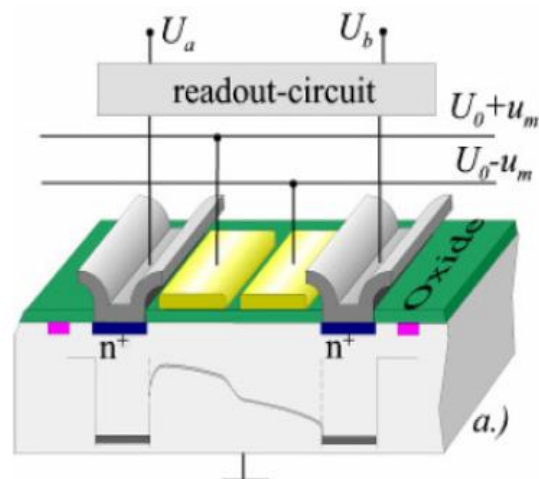
Esto puede solventarse con una buena sincronización entre la luz emitida por la parte activa del sensor y la temporización de captura de la cámara, evitando así que luz emitida de otras capturas llegue en los tiempos que no le corresponde. [17]

Este error se clasifica como error sistemático y tiene soluciones más o menos sencillas y de manera permanente, sin embargo, falta hacer mención a otros posibles errores que no son sistemáticos, como por ejemplo el movimiento de un objeto. Estos errores no sistemáticos tienen soluciones más complejas y que en general son mucho más costosos de solucionar.

En este documento solo mencionaremos uno de los errores más llamativos, el ‘motion blur’. Este error es la deformación de la imagen capturada propiciada por el movimiento del objeto, si miramos la imagen siguiente podremos ver como la silla de la imagen se ven deformadas, o con cierto error de captura debidos a un movimiento de la misma.



27. Imagen resultado de varias capturas a largo de un objeto en movimiento. [14]



28. Pixel ToF

En cuanto a la forma de captura del sistema tradicionalmente han existido de forma genérica dos formas de capturar la luz, ‘global shutter’ y ‘rolling shutter’. ‘Rolling shutter’ consta de capturar la imagen fila a fila consecutivamente, por otro lado, el ‘global shutter’ se basa en capturar la luz en todos los píxeles al mismo tiempo, como el lector intuirá durante el ‘global shutter’ hay un consumo muy superior con respecto a ‘rolling

shutter’ dado que todos los píxeles actúan al mismo tiempo. Esto es cierto, sin embargo, si hubiera algún objeto moviéndose a velocidades cercanas a la consecución de las filas al capturar, se pueden obtener imágenes deformadas de la realidad, pues habrá grandes diferencias en la escena entre la captura de la primera fila y la última. Puesto que además en los sistemas de **ToF** la sincronización del tren de pulsos es vital, se usará el método de captura ‘global shutter’.

4.2 Aplicaciones.

Este tipo de sensor nos da unas posibilidades enormes teniendo en cuenta las aplicaciones en las que pueden ser utilizadas. Impresión 3D, sistemas autónomos, seguridad, control industrial, biométrica, etc. [17] Gracias a las características de estos sensores es interesante usarlo en numerosas aplicaciones pues nos permite extraer propiedades geométricas de los objetos.

Estos sensores se han utilizado en aplicaciones de robots móviles aun teniendo en cuenta incluso posibles problemas que pudieran surgir debido a una resolución limitada. Gracias a la nube de puntos se pueden procesar los datos con una variante del algoritmo ‘Iterative Closest Point’ (**ICP**) para relacionar dos puntos de la nube. Por ejemplo K. Ohno lo usa en una de sus bibliografías para guiar un robot que imita a una serpiente para aplicaciones de rescate en escenarios de desastre.

4.3 Sensores en ToF: Elección de pixel.

Como se ha explicado anteriormente, el pixel 9T presenta ciertas ventajas que pueden ser explotadas en aplicaciones de ToF, si bien no es necesario utilizar una segunda rama, y que al hacerlo reducimos el ‘fill factor’, la ventaja de poder utilizar dos capturas muy cercas en el tiempo no puede desecharse.

Al utilizar estas dos ramas podremos capturar y comparar en un periodo de tiempo muy pequeño. En lo que queda de documento nos centraremos en el pixel 9T para llevar a cabo nuestra implementación. A continuación, se volverá a explicar de manera más concisa el funcionamiento del pixel para que no haya lugar a dudas.

Siendo PPD el fotodiodo encargado de generar la carga, AB será una señal a cargo de conectar dicho fotodiodo a una tensión alta, consiguiendo limpiar la carga que pudiera haber almacenada en el pixel de una captura a otra, esto evita que se lea carga de una captura anterior a la del momento actual.

Con la señal TX controlamos la integración o no de la carga que esté almacenada en PPD gracias al condensador FD. Del mismo modo que limpiábamos la carga en el fotodiodo con la señal AB, la señal RST nos permitirá descargar la tensión almacenada en el condensador FD, limpiando cualquier residuo de una captura anterior.

El terminal del condensador FD, se encuentra conectado a un seguidor de fuente que nos permite leer la tensión almacenada sin afectarla, modificarla o destruirla durante la lectura. Finalmente, con la señal SEL, podemos conectar el seguidor de fuente a la salida. Ésta será una línea de alta impedancia, al activar la señal SEL la línea de salida se verá automáticamente ocupada por la lectura del pixel. La función de esta señal es obtener una salida de la lectura de los píxeles controlada.

5 DISEÑO DE BÚFERES DE SEÑALES DE CONTROL DE UN PIXEL PARA TOF: CÁLCULOS TEÓRICOS

The real problem is not whether machines think but whether men do.

- B.F. Skinner -

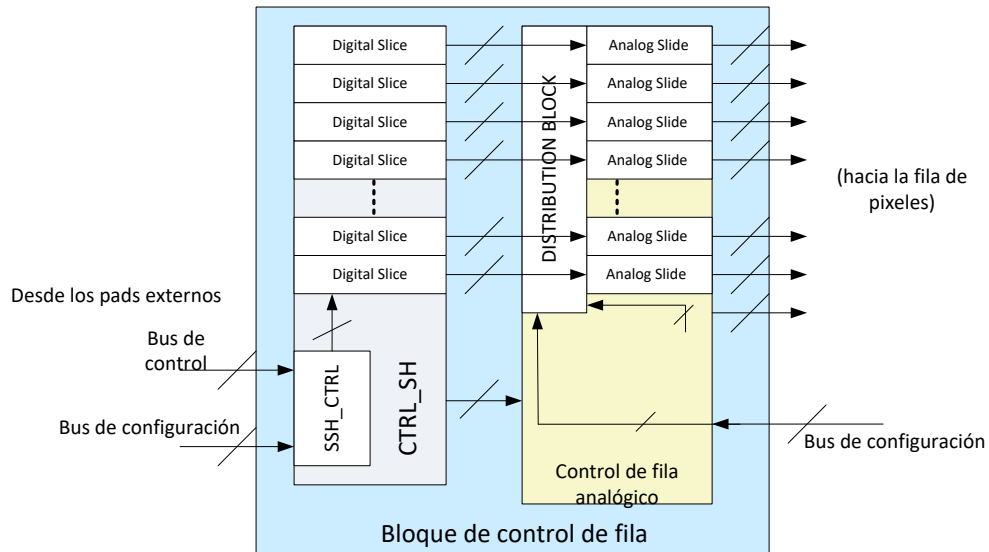
Profundizaremos en este apartado en los cálculos teóricos para el diseño de todo el ‘buffer’ para el control de fila. Se presentará el bloque del control de fila desde un punto de vista global hacia una vista más detallada de cada componente. A lo largo de este capítulo se presentan varias topologías para desempeñar la misma función desde topologías más básicas a versiones más complejas de las mismas que garantizan mayor control sobre las señales a tratar. Si bien estos circuitos constituyen una parte muy importante de este documento, también se detallará el impacto del tamaño y forma del pixel al que los ‘buffers’ atacan pues supondrá en última instancia un limitante inevitable.

El simple hecho de utilizar una circuitería supondrá una serie de retrasos y limitaciones, con el objetivo de desarrollar un buen diseño procederemos a un estudio preliminar de las consecuencias que conllevan utilizar cierto tipo de tecnologías e incluso del uso del pixel en sí.

Conforme la complejidad de nuestros circuitos vaya aumentando, el área utilizada para el control de fila se verá afectado, pero además nuestros tiempos de respuesta pueden verse igualmente alterados. Nuestro objetivo será conseguir un diseño versátil de cierta complejidad que no afecte a los tiempos de respuesta más allá de lo que un diseño simple lo hace.

5.1 Bloque de Control de fila.

Esta arquitectura se encarga del control de las filas de pixeles, está compuesto por dos sub-bloques bien diferenciados. Por un lado está el control de bloques digital SSH (Sensor Shifter) y un bloque analógico que lo sigue llamado control analógico de fila. Este bloque opera como una interfaz entre los controles digitales y la fila de pixeles operando, por tanto, como ‘buffer’. En la siguiente imagen podemos observar cómo se distribuyen estos dos bloques en el bloque de control de fila.



29. Esquema del bloque de control de fila.

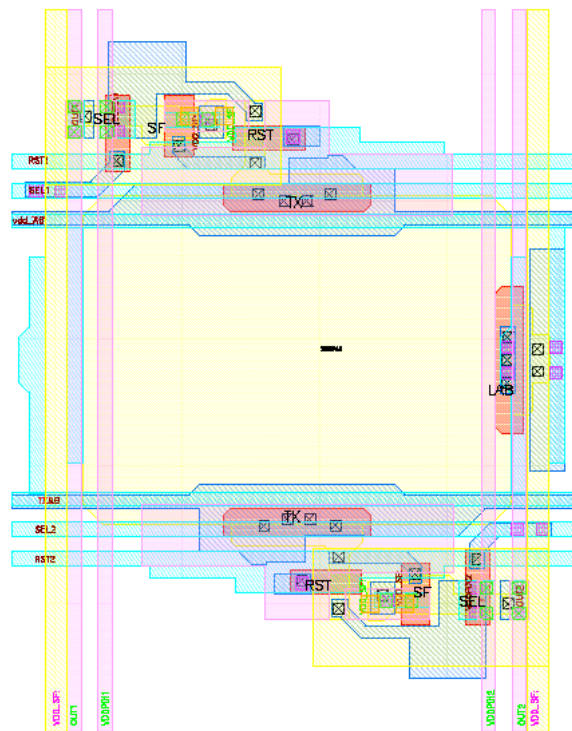
En este documento nos centraremos en el control de fila analógico. Las formas de onda que controlan el pixel se generan externamente durante la exposición. Las señales **SEL** y **RST** operan durante el proceso de lectura y vienen dadas también de un bloque externo. Todas estas señales externas llegan al bloque **SSH_CTRL** que las distribuye a distintos ‘slices’ según correspondan y cada uno de estas filas se ocupará de un pixel en concreto.

Como ya se ha comentado antes, la interfaz analógica, actúa como un buffer, además esta cambiará los niveles lógicos de la parte digital a los niveles adecuados de tensión para cada una de las partes del pixel.

Esto incluye a todas las señales de control, que dispondrán de distintos niveles de tensión adecuados al pixel, más adelante se profundizará en la posibilidad de establecer estos niveles como programables para realizar un bloque reutilizable para varios tipos de píxeles. Además de la posibilidad de programar los niveles de tensión de alimentación, se explorará la posibilidad de programar la pendiente de subida de las señales que lo atraviesen, con esto podríamos conseguir adaptar el ‘buffer’ para distintos tipos de operaciones sin tener que recurrir a otros diseños para el sensor.

5.1.1 Pixel array: Layout del pixel y extracción.

Centrados nuestros esfuerzos en el pixel 9T, podemos empezar a hacer cálculos acerca de sus parásitos, etc. Se tomará como muestra un que ya ha sido probado anteriormente, la disposición física de este sensor es la mostrada en la imagen inferior.



30. Layout del pixel elegido.

Las características de este sensor, son 0.285 μm de ancho de pista y 10 μm de largo de la misma. Esta información nos es muy relevante, pues dependiendo del tamaño de la pista, el ancho o el solapamiento con otras pistas, nuevas capacidades parásitas podrán aparecer.

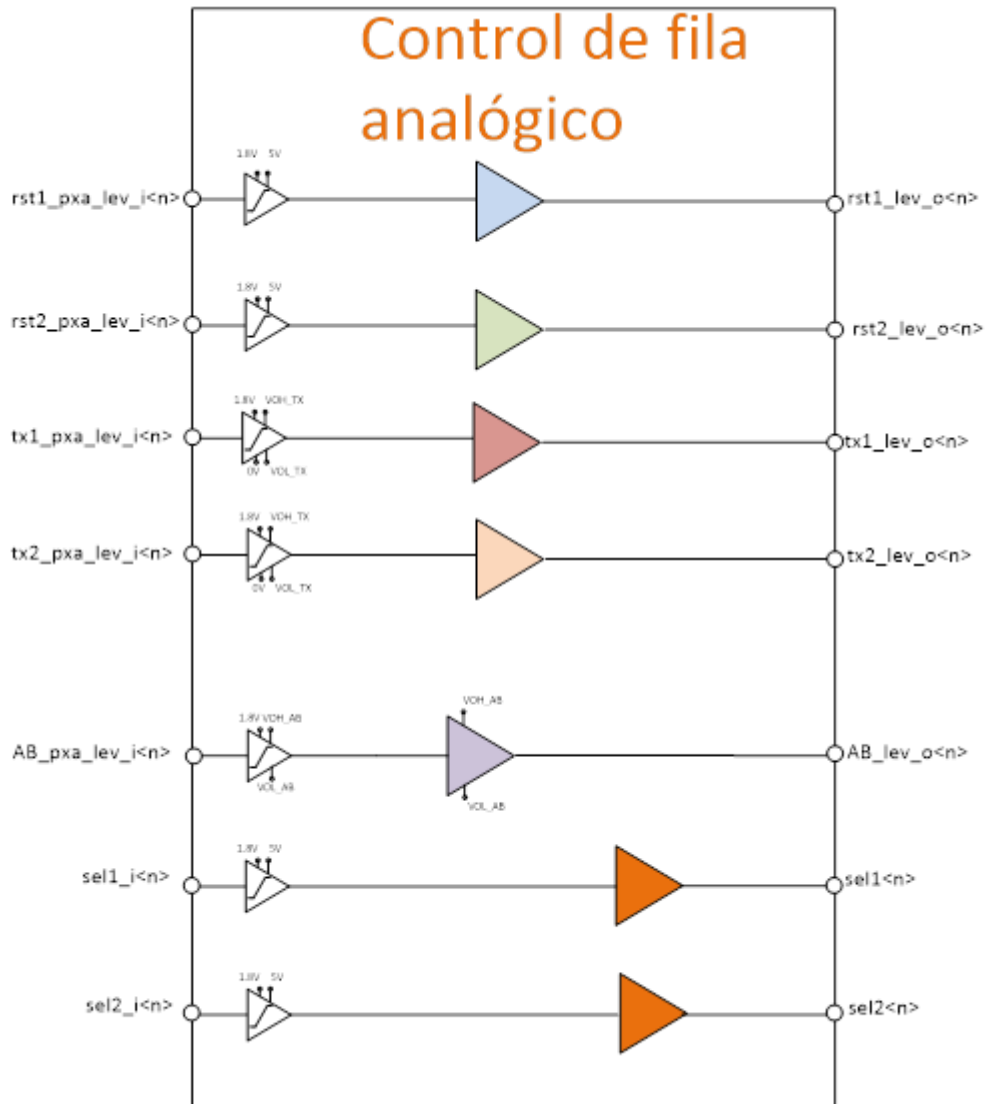
Para realizar el cálculo de parásitos en la línea será necesario primero asumir ciertas aproximaciones, que serán lo suficientemente exactas para no alejar nuestros cálculos de la realidad. Tras algunas pruebas de extraído con el sensor, y corroboraciones de los cálculos, en los que no es necesario profundizar, se han llegado a las siguientes aproximaciones:

-Capacidad de óxido: 3 fF/m^2 para la señal de transferencia y antiblooming, 2.5 fF/m^2 para las demás.

-Capacidad de 'overlap': 0.35 fF/m^2 .

5.2 Control de fila analógico.

Volviendo a un nivel de jerarquía superior, ubicaremos al lector sobre la disposición del sensor completo. Dentro del bloque de la figura 5.1, el bloque de control de fila analógico, quedará definido como se muestra en la siguiente imagen. Este bloque será el encargado de transportar la señal hasta el pixel además de adecuarla debidamente al dominio analógico.



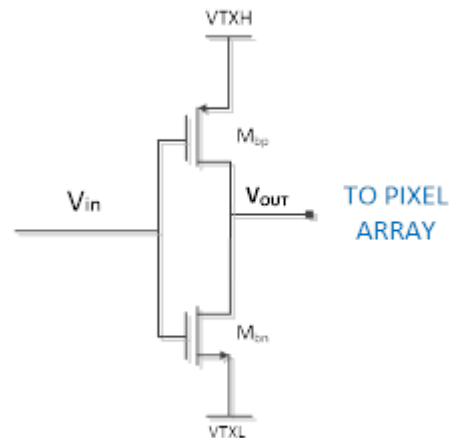
31. Bloque de control de fila analógico. Vista de cada slice.

El bloque representado en la anterior figura no es más que una de tantas que se encargarán de cada uno de los ‘slices’. Dentro de un solo ‘slice’, habrá por tanto uno solo de estos bloques. Dentro de este bloque, por cada señal habrá un ‘level shifter’, que es el encargado de cambiar la señal del dominio digital al dominio analógico. Tras esta conversión de dominio, cada una de las señales atravesará un ‘buffer’ que se encarga, no solo de regenerar la señal sino también de proveer una salida capaz de soportar el viaje por la línea del sensor, de otro modo la señal se degeneraría hasta quedar irreconocible o imperceptible. Por tanto cada señal tras atravesar su ‘buffer’ saldrá del bloque directamente hacia su pista correspondiente del pixel. Es necesario recordar, que cada pixel mostrado en la figura 5.2 estará conectado a otro igual en cada dirección, creando una matriz del sensor, y además las líneas de transmisión por donde nuestras señales de control se moverán hasta el pixel designado.

Ahora profundizaremos en cada uno de los bloques electrónicos que conforman nuestro control de fila analógico.

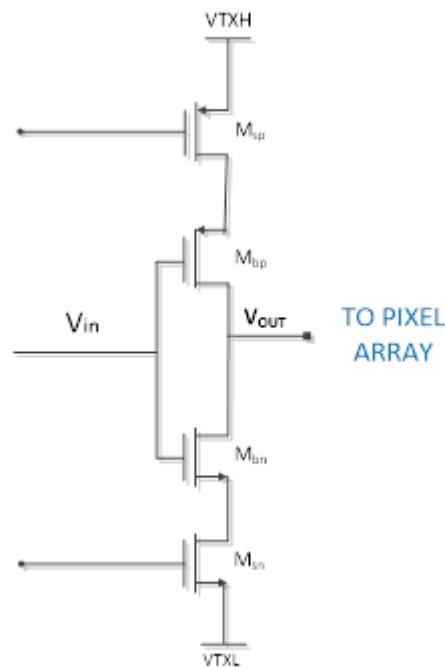
5.2.1 Buffers

Comenzamos con el diseño del ‘buffer’ que será el alma del control de fila analógico. Puesto que el control de fila analógico no es más, en resumidas cuentas, un ‘buffer’, el buen diseño de este bloque será crucial para un buen comportamiento del conjunto. El objetivo al diseñar este bloque será que el impacto de añadir estos semiconductores sea mínimo, y todos los retrasos o alteraciones de las señales queden en términos de parámetros no resumibles. Sin embargo, en la realidad este paso se resumirá en un ‘trade-off’, donde deberemos elegir hasta qué punto podemos asumir ciertos aspectos no deseables, para poder conseguir una señal más limpia. Un aspecto importante que se pretende remarcar en este documento es la necesidad o beneficio de la complejidad de los circuitos. Por ellos comenzaré el diseño de varios ‘buffers’ desde más simple a más complejo, para más tarde poder comparar sus resultados y comprobar el impacto de añadir complejidad al diseño. El primer diseño será, por tanto el ‘buffer’ más simple posible, un inversor.



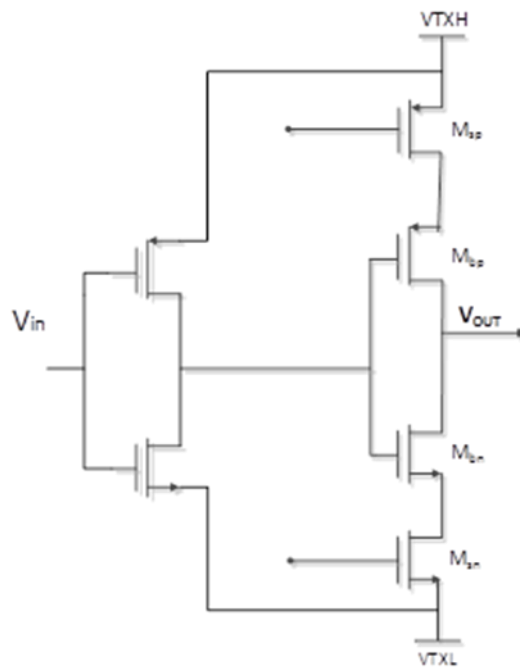
32. Buffer simple sin control de slew.

La necesidad de controlar la rampa de salida del buffer, nos exige que añadamos complejidad al diseño, de este modo, al añadir un par de transistores, seremos capaces de controlar el slew de nuestras señales cambiantes gracias a la conexión de una tensión en la puerta de los mismos. El siguiente paso en el diseño será este esquemático:



33. Buffer con control de slew.

Sin embargo la funcionalidad del diseño no es del todo deseable, pues nuestra señal saldrá invertida. Añadir una pre-etapa al diseño no solo nos permitirá mantener la señal sin invertir, sino además escalar la carga capacitiva de entrada, de manera que el cambio en la señal no será abrupto, retrasando mínimamente la señal. Nuestro diseño de buffer final será con el mostrado en la siguiente figura:



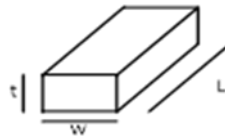
34 . Buffer con control de slew y preetapa.

5.2.1.1 Carga de salida.

El primer paso en el diseño del 'buffer' es conocer la carga que habrá a la salida, es decir, la carga parásita que se encuentra en las pistas de transmisión. Las pistas de transmisión se recuerda que no son más que las líneas pertenecientes a cada pixel. Nuestro primer objetivo será entonces, calcular las capacidades y resistencias parásitas que se encuentran en el pixel y por tanto en la línea de pixeles.

Empezaremos analizando de donde pueden proceder los parásitos que entorpecerán a las señales que recorran la línea. Es necesario recordar que habrá cambios bruscos en el potencial de las señales, por tanto una de las mayores problemáticas que nos encontraremos a la hora de los parásitos serán los retrasos. Una capacidad parásita muy grande, por ejemplo, puede ralentizar la subida del potencial de una señal, que de manera ideal sube inmediatamente.

Comenzaremos el análisis aislando la pista de un solo pixel de manera genérica, de esta forma podremos extrapolar los resultados a cualquier resolución de sensor y para cuantos pixeles queramos en la fila. Llamaremos R_{unit} y C_{unit} a la resistencia y capacidad parásita respectivamente de un solo pixel para, posteriormente poder multiplicarlo por el tamaño de fila deseado. R_{unit} y C_{unit} dependerán de la forma, longitud, tamaño y posicionamiento de la pista de control.



35. Representación de una pista. W ancho, L largo y t espesor.

$$R = \rho \cdot \frac{L}{W \cdot t}$$

ρ : Resistividad del metal por cuadrado.

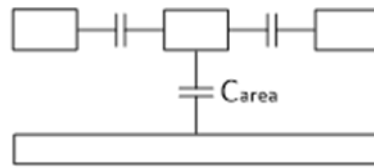
L: Longitud de la pista

W: Ancho de la pista.

T: espesor de la pista.

En este trabajo obviaremos t ; el grosor de la pista, ya que tendrá poco efecto en el cómputo total. Para la capacidad en cambio tendremos que tener en cuenta dos tipos de capacidad, la intrínseca y la extrínseca, debida a la conexión a puerta de los transistores y a la línea de control respectivamente.

Para el cálculo de la capacidad extrínseca, partimos del tamaño de la pista y lo multiplicaremos por la capacidad por metal. Además se tendrá en cuenta la capacidad creada por los laterales de la pista. Sabemos que la capacidad con respecto a sustrato será proporcional al área que proyecta sobre él y la lateral proporcional a la longitud que comparten e inversamente proporcional a la distancia que los separa.

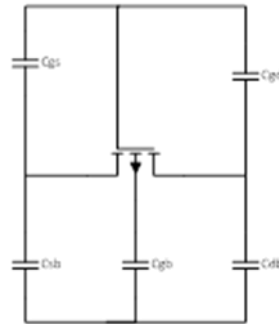


36. Modelo de capacidades parásitas de línea.

$$C_{ext} = K_{area} \cdot W \cdot L + K_l \cdot \frac{L}{S}$$

Donde K es una constante que determina la capacidad unitaria y S la distancia entre pistas.

Por otro lado para la capacidad intrínseca se usará el siguiente modelo:



37. Modelo de capacidades parásitas del mosfet. [20]

Se usará la aproximación $\frac{1}{2} \times W \times L \times C_{ox} \times C_{ov}$ debido a la forma de los transistores en el pixel.

Tenemos pues que la resistencia y la capacidad unitaria serán:

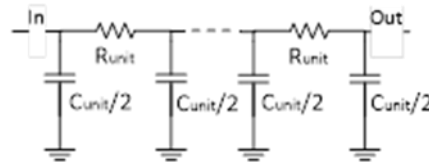
$$C_{unit} = \frac{1}{2} \cdot W_{mos} \cdot L_{mos} \cdot C_{ox} + C_{ov} + K_{area} \cdot W \cdot L + K_l \cdot \frac{L}{S}$$

$$R_{unit} = \rho \cdot \frac{L}{W}$$

Utilizadas las fórmulas, obtenemos los siguientes valores unitarios y los valores aportados por el pixel, obtenemos los siguientes valores:

R: 5.26 Ohm C: 3.338 fF

Ahora que sabemos que la línea estará compuesta por resistencias y capacidades parásitas he decidido hacer uso de un modelo π como el de la figura.



38. Modelo pi de la línea.

Se usará un modelo distribuido para acercarnos más a la realidad, R_{unit} será la resistencia asociada a la línea de un solo pixel, y C_{unit} su homólogo para la capacidad. Ahora que conocemos cual será el modelo unitario vamos a generalizar para una resolución de tamaño N.

Los valores totales de los parásitos a lo largo de una línea de N píxeles serán:

$$R_t = N \cdot R_{unit}$$

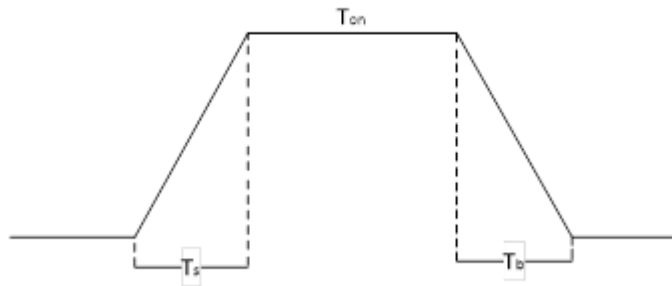
$$C_t = 2 \cdot \frac{C_{unit}}{2} = N \cdot C_{unit}$$

Todos los cálculos y el diseño del control de fila se basarán en el tamaño de 1280 columnas. Esto quiere decir que la resistencia total y la capacidad total de la línea quedan como siguen:

R: 6.73 KOhm C: 4.272 pF

Será interesante además ver en que afecta esta carga de salida independientemente del circuito que tengamos atacándolo. Dada la tecnología que estamos usando en este documento un dato interesante para analizar es la velocidad, como hablamos de parásitos y de señales con forma de tren de pulso, una buena medida es el tiempo de subida de las mismas.

A lo largo de todo el recorrido la señal se verá sometida a distintos efectos por los que acabará con un slew inadecuado. En la figura se puede observar la señal de control que se usa normalmente en este tipo de aplicación y como se ve afectada por los parásitos.



39. Representación de señal de control.

Teniendo esto en cuenta tendremos que centrarnos en las columnas del pixel que se ha mostrado en apartados anteriores. Nuestro esfuerzo se centrará en hallar cual puede ser nuestra velocidad teórica máxima para las

configuraciones que se han ofrecido al principio de este documento. Para ello hallaré cual será nuestro tiempo de ‘setling’ de manera teórica basándome en todo lo calculado hasta el momento.

Se sabe que el potencial en una conmutación no instantánea evoluciona de la siguiente forma:

$$V(t) = V_o \cdot e^{\frac{-t}{\tau}} \quad (5)$$

siendo $\tau=RC$

Debido a los bits de nuestro bloque digital habrá un cierto nivel de precisión a la hora de enviar un pulso en el nivel correspondiente alto. En cuanto a la precisión de nuestro cálculo para el ‘setling’, dado esta incertidumbre introducida por el LSB, no tiene sentido que busquemos un grado mayor de precisión en nuestros cálculos. Dicho lo cual y conociendo la ecuación (5) deducimos que lo que buscamos es; siendo B el n° de bits del bloque digital.

$$e^{\frac{-N \cdot \tau}{\tau}} = \frac{1}{2^B}$$

Ya que la parte correspondiente a la exponencial determinará la relación entre $V(\tau)$ y V_o , de esta manera nos aseguramos que el porcentaje de error se encuentra dentro de los márgenes del LSB. Se extrae de esta ecuación que a menor n° de bits más rápido podremos permitirnos ir.

Aunque B vendrá dado por nuestro generador vamos a considerarlo un parámetro por el momento pues será lo que fije nuestra precisión. Con el estudio realizado hasta el momento podemos asegurar que:

$$V_f = V_o \cdot e^{\frac{-\ln(1/2^B)}{\tau}}$$

$$t_{set} = \ln\left(\frac{1}{2^B}\right) \cdot \frac{N^2}{2} \cdot R_{unit} \cdot C_{unit} = f(B, N, R, C)$$

Por tanto el tiempo de ‘setling’ con todos los valores que hemos calculado hasta ahora será:

$$t_{set} = \ln\left(\frac{1}{2^B}\right) \cdot \frac{N^2}{2} \cdot \left(\rho \cdot \frac{L}{W}\right) \cdot \left(\frac{1}{2} \cdot W \cdot L \cdot C_{ox} + C_{ov} + K_{area} \cdot W \cdot L + K_l \cdot \frac{L}{S}\right) = f(B, N, R, C)$$

Se ve a primera vista que el papel más importante lo juega N, n° de píxeles, sin embargo, ρ y L también aportan directamente retraso, en realidad ρ depende exclusivamente de la tecnología y por tanto no podrá ser considerado un parámetro de diseño, pero L sí. L nos designará cual será el largo de la fila de píxeles.

W por su parte puede mejorar el tiempo de ‘setling’ aunque su impacto dependerá de la relevancia de C_{ov} y aun así nos exigirá un compromiso, pues a mayor tamaño menor área efectiva habrá en el píxel.

Una vez hemos hallado de que depende el tiempo de 'setling', vamos a discernir cuales son variables que podemos alterar según nuestro diseño. Las constantes y características de nuestra ecuación serán C_{ox} , C_{ov} , ρ , K_a , K_l , que dependerán exclusivamente del tipo de tecnología que estemos usando, por lo tanto no pueden ser parámetros de diseño.

En cuanto B, el número de bits, puede afectar al tiempo de subida, aunque realmente afecta a la precisión, con lo que si elegimos un tamaño de bits muy pequeño nuestra precisión será muy baja y puede que no aseguremos que los transistores funcionen correctamente. Por lo tanto B puede afectar a nuestro diseño pero en un rango muy ajustado y de una forma virtual. De este modo la mejor opción es fijar un nivel de precisión aceptable e intentar diseñar con los demás parámetros

Por otro lado N; el número de pixeles, sí que es un parámetro de diseño, que además, posee un gran impacto en la velocidad de nuestro circuito, hay que tener en cuenta que no es un parámetro minimizable, pues mientras más pequeño mejor, pero habrá que llegar a un compromiso entre la resolución que se puede aceptar por ganar velocidad.

Si nos fijamos en W; el ancho de la pista, aunque también es un parámetro de diseño tampoco se puede minimizar, su tamaño ideal sería lo más grande posible, pero esto empeoraría el porcentaje de área activa del pixel. De nuevo habrá que alcanzar un compromiso entre el ancho de la pista y el área efectiva.

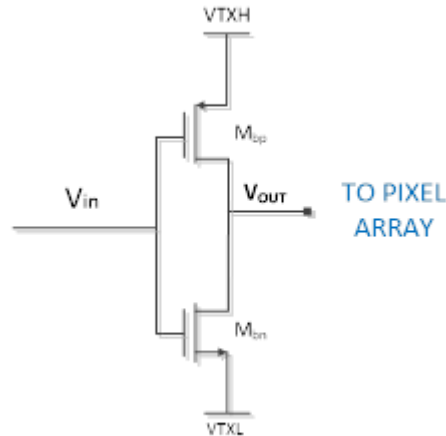
También S; el espacio entre pistas de control, será un parámetro de diseño, pero contribuye poco al tiempo y además aumentar S para bajar el tiempo de subida implica empeorar el área efectiva. Habrá que lograr encontrar de nuevo un punto medio entre disminuir el tiempo y no perder demasiada área efectiva.

Por último L; la longitud de la pista, y por tanto el largo del pixel, también será un parámetro de diseño que a su vez será minimizable pero que sus extremos dependerán exclusivamente de W, S y constantes. Sin embargo aumentar L tiene varias consecuencias. Si el pixel es muy estrecho, para lograr tener mayor área sensora habrá que tener una N mayor. También será un parámetro clave en nuestro diseño.

En definitiva, en primera aproximación nuestro tiempo de subida será directamente proporcional al tamaño del array (N y L) e indirectamente proporcional al ancho de la pista (W) y el espacio entre ellas (S).

5.2.1.2 Buffer sin control de slew.

Comenzaremos con el diseño de un buffer sin control de slew, su esquema es el que se muestra en la siguiente imagen.



40. Etapa de salida del buffer sin control de slew.

Comenzaremos con un pequeño análisis de cómo funciona el circuito. Si V_{in} se encontrara a V_{TXH} el transistor M_{bp} se encontraría en corte, pues no habría tensión V_{gs} . Por el contrario el transistor M_{bn} se encontraría en zona lineal de modo que operaría como una resistencia.

Como hemos estudiado antes, el retraso en la subida del inversor dependerá de la carga a la salida del mismo, de manera que los cambios abruptos evolucionarán de forma exponencial.

No podemos perder de vista que este inversor irá conectado a línea de control, de manera que la carga vista desde su salida será la de la línea más la resistencia del MOS correspondiente dependiendo si está en óhmica el transistor p o n. De este modo a la hora de diseñar este inversor tendremos dos objetivos. Por un lado la resistencia en óhmica del transistor p y la resistencia del transistor n deberán ser lo más parecidas posibles para que los tiempos de subida y de bajada estén en órdenes parecidos. Por otro lado buscaremos que lo que domine la dinámica no sea el inversor, sino la línea, puesto que la línea es inamovible y si la dinámica la dominara el inversor, significaría que aún podemos ganar en velocidad.

Se comienza el estudio partiendo del supuesto de que el transistor n este en zona lineal y el transistor p se encuentre en corte. Es decir a la entrada tenemos un nivel alto lógico y a la salida un nivel bajo.

Teniendo esto en cuenta, seguiremos el siguiente razonamiento:

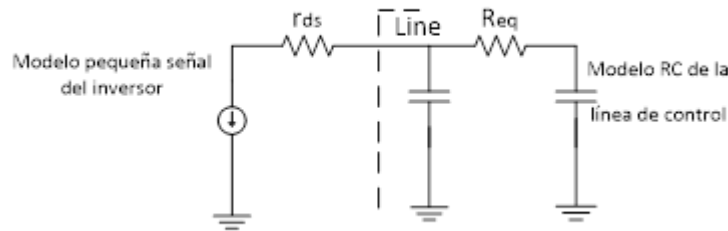
$$i_{ds} = k_n \cdot \frac{W_n}{L_n} \cdot [(V_{gs} - V_T)V_{DS} + \frac{V_{DS}^2}{2}]$$

Puesto que estamos en óhmica, es fácil llegar a la conclusión de que:

$$r_{ds} = \frac{v_{ds}}{i_{ds}} = \frac{1}{K_n \cdot \frac{W_n}{L_n} \cdot (V_{gs} - V_T)}$$

Una vez que ya sabemos cuál es el valor de la resistencia aportada por el inversor, ahora deberíamos intentar mirar esto en el cómputo general, de manera que veamos en que puede afectar esta resistencia con el conjunto del control de fila.

En definitiva, en estos casos estaremos operando con lo siguiente:



41. Modelo etapa de salida + modelo pi.

Si observamos el modelo completo y atendemos a uno de los objetivos que nos marcamos en un principio, tendremos que diseñar nuestro inversor de manera que r_{ds} no contribuya a la R total, de modo que esto debe ser $r_{ds} \ll R_{eq}$.

Sabemos que la R_{eq} para todas las señales es igual a $6.73 \text{ K}\Omega$, r_{ds} debe de encontrarse al menos un orden de magnitud por debajo. Escogeremos pues 3 órdenes de magnitud menor: $r_{ds} = 0.6 \text{ K}\Omega$.

Por otro lado para lograr que los tiempos sean aproximadamente los mismos, tenemos que fijarnos en que pese a que el análisis del otro transistor sea análogo, K_n y K_p no son iguales, y dependen de la tecnología, sin embargo, por lo demás todo es igual, de modo que conseguir la misma resistencia o al menos una muy parecida es tan sencillo como escalar el tamaño del transistor p de modo que las variables K se igualen, es decir:

Sabemos que típicamente:

$$K_p = 3.33 \cdot K_n(1)$$

Por tanto podemos hacer:

$$K_p \cdot \frac{W_p}{L_p} = K_n \cdot \frac{W_n}{L_n}$$

Usando la ecuación (1):

$$3.33 \cdot K_n \cdot \frac{W_p}{L_p} = K_n \cdot \frac{W_n}{L_n}$$

Por tanto:

$$3.33 \cdot \frac{W_p}{L_p} = \frac{W_n}{L_n}$$

Con estas dos conclusiones ya podemos realizar nuestro primer inversor que se ajuste a las características que

buscamos y que además, al depender solo de la resistencia, nos servirá como primera base para el diseño de todas las señales.

Aplicando el valor escogido de r_{ds} y utilizando la siguiente ecuación:

$$r_{ds} = 0.2K\Omega = \frac{1}{200 \cdot \frac{W_n}{L_n} \cdot (5 - 0.7)}$$

$$\frac{W_n}{L_n} = 1.93 \approx 2$$

Utilizando ahora la ecuación (1):

$$\frac{W_p}{L_p} = 3.33 \frac{W_n}{L_n} = 6.67$$

Para facilitar en el futuro una más fácil implementación del layout vamos a fijar $L_n = L_p = 0.6$.

Con esto $W_n = 1.2$ y $W_p = 4$

Ahora que ya poseemos un inversor desde el que partir, el siguiente paso es formar un buffer. Para ello necesitaremos crear otra etapa en la que se invierte la señal con anterioridad, creando una doble inversión y por lo tanto restaurando la señal original. Al hacer esto, hay que tener en cuenta que debemos hacer un cambio gradual en las capacidades de salida, de manera que, de etapa a etapa, logremos un aumento relativo de las cargas a la salida.

Vamos a utilizar la siguiente ecuación para lograrlo:

$$Me^n$$

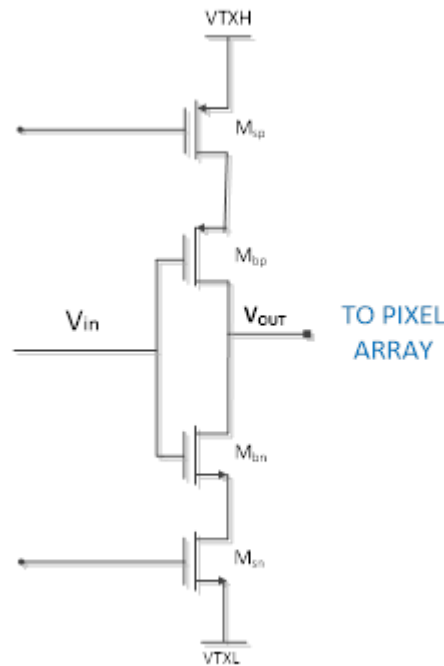
Donde M será el tamaño del mos, y n el número de la etapa, como nuestro buffer constará de dos etapas, la preetapa, será con $n=0$ y nuestra tetapa de salida será $n=1$, solo basta con despejar para clarificar cuales son los tamaños básicos de los mos de la preetapa. Realizando los cálculos obtenemos:

$$M_p = 1.5$$

$$M_n = 0.6$$

5.2.1.3 Buffer con control de slew.

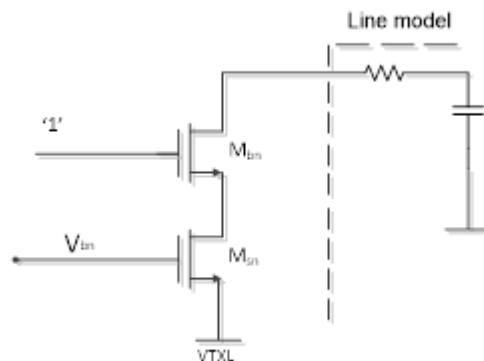
Con la siguiente configuración buscamos poder controlar el tiempo de subida de las señales de control.



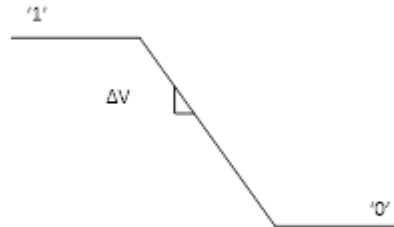
42. Etapa de salida del buffer con control de slew.

Con esta configuración logramos controlar el slew de la señal de salida gracias a lo cual conseguiremos limitar la corriente de pico consumida. Realizaremos un pequeño estudio de la topología. Examinaremos en concreto el caso en el que la entrada es un '1' lógico y la salida un '0', además tenemos que tener en cuenta que esta topología irá conectada por su salida a un modelo RC que representará la línea de transferencia.

Analizaremos la transición del '1' al '0' lógico. Tenemos que tener en cuenta que el transistor Msn esta pensado para actuar como una fuente de corriente controlada, de modo que se pretende que trabaje en saturación. Si realizamos un pequeño análisis en gran señal primero observaremos lo siguiente:



43. Modelo de la etapa de salida del buffer con control de slew.



44. Rampa de bajada de nivel alto a nivel bajo en la línea de transmisión.

Si nos centramos en la carga y descarga del condensador del modelo de la línea:

$$I = C \cdot \frac{dV}{dt}$$

Y por tanto:

$$SlewRate = \frac{\Delta V}{\Delta t} = \frac{I}{C}$$

Esto nos indica que efectivamente, si controlamos la intensidad con el nuevo mos que hemos incluido podremos controlar el SlewRate de la salida, puesto que ya conocemos la capacidad de la línea, los tiempos en los que pretendemos controlar el slew rate y la diferencia de potencia en el que se da este slew, podemos calcular directamente la corriente, y con ella, el tamaño de nuestros transistores:

$$I_{dc} = 0.6572 \text{ mA}$$

Sabiendo que deberán estar actuando en saturación hacemos uso de la ecuación de saturación del transistor:

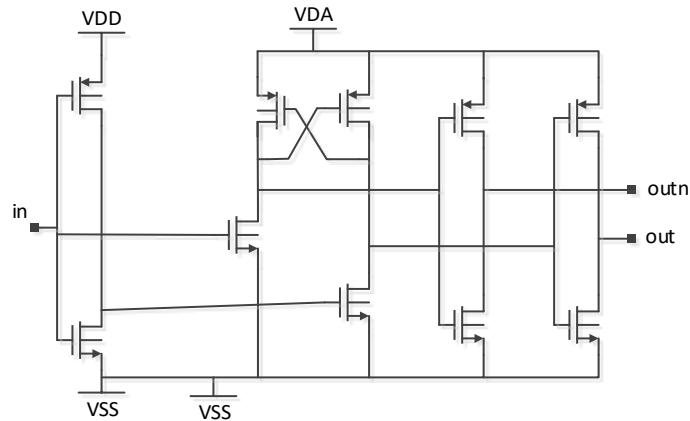
$$I_{dc} = \frac{K}{2} \cdot \left(\frac{W}{L}\right) (V_{dsat})^2 \Rightarrow \left(\frac{W}{L}\right) = 105 \Rightarrow W = 63.09 \sim 64; L = 0.6$$

Para el transistor tipo p se utiliza la misma ecuación obteniendo un tamaño W de 104 micras.

5.2.2 Level shifters

5.2.2.1 Parte analógica y digital

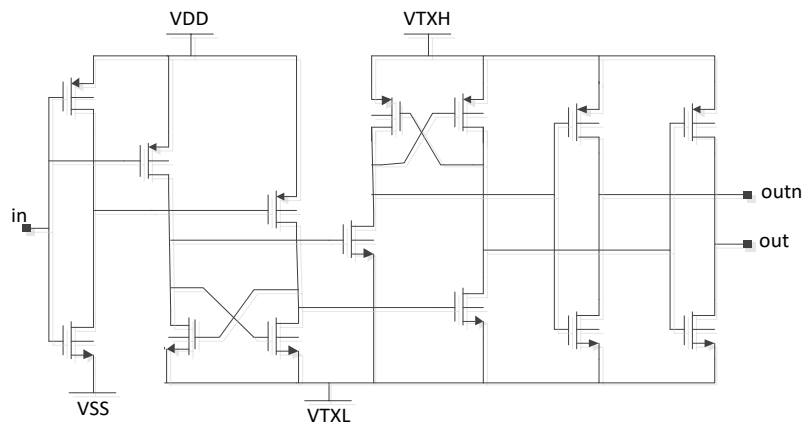
El level shifter consta de dos partes, por un lado una parte dedicada a la tecnología digital y por otra parte a la tecnología analógica.



45. Level shifter con nivel alto programable y tierra común.

En la imagen anterior podemos ver el inversor destinado al dominio digital. El resto de la tecnología es perteneciente a la parte analógica. Entre estas dos etapas se encuentra el corazón del level shifter, estos transistores se encargarán de convertir los niveles digitales a niveles analógicos.

Nuestro dispositivo más simple se encargará de subir los niveles altos del dominio digital al analógico manteniendo una tierra común. Sin embargo esta no es la única forma de proceder, pues podemos además añadir una etapa extra para poder separar la tierra digital de la analógica, consiguiendo a su vez que los niveles de tensión altos y bajos del dominio analógico sean completamente programables.

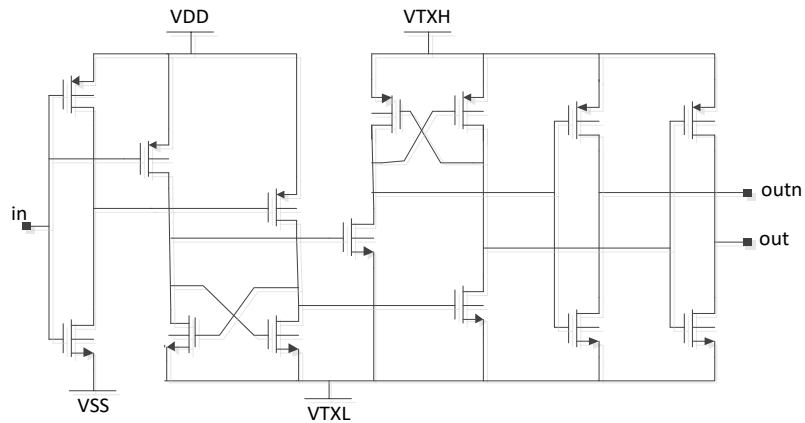


46. Level shifter con nivel alto y bajo programables.

Si bien poseer unos niveles altos y bajos programables es una gran ventaja, esto trae consigo una serie de desventajas. Aunque a primera vista en el diseño pueda parecer un ligero cambio, hacer este cambio nos forzará a aislar los dos dominios con la idea de que no se “ensucien” entre ellos. La complejidad de la disposición física de los metales se verá profundamente afectada, pues además deberemos salvar una distancia entre los dominios. En el apartado de implementación podremos ver el resultado final de esta separación.

5.2.2.2 Level shifter

Este apartado se dedicará a los cálculos para el diseño de un level shifter clásico, sin tensiones programables. El esquema es el que se muestra en la siguiente imagen.



Como se puede observar se ha optado por hacer un diseño genérico para la posible reutilización del bloque.

Comenzaremos con el diseño de las dos etapas de salida, en principio son etapas idénticas.

Si nos remontamos a la imagen (32) observamos que el level shifter irá directamente acoplado a la entrada de nuestro buffer. Si hacemos observamos con más detenimiento los diseños de ambos bloques podemos observar que se tratan de etapas de inversores uno tras de otro. De esta manera podemos seguir la misma regla de etapas, pues nuestro interés residirá en que las etapas anteriores no afecten a las etapas posteriores. Si volvemos a aplicar las mismas fórmulas que para la pre-etapa del buffer, podemos construir directamente los tamaños del inversor de salida del level shifter.

De este modo nuestra primera etapa (o etapa 0) será la salida del level shifter, nuestra siguiente etapa (o etapa 1) se corresponderá con la pre-etapa del buffer y por último la etapa de salida del buffer será la etapa 2. Aplicando la fórmula:

$$Me^n$$

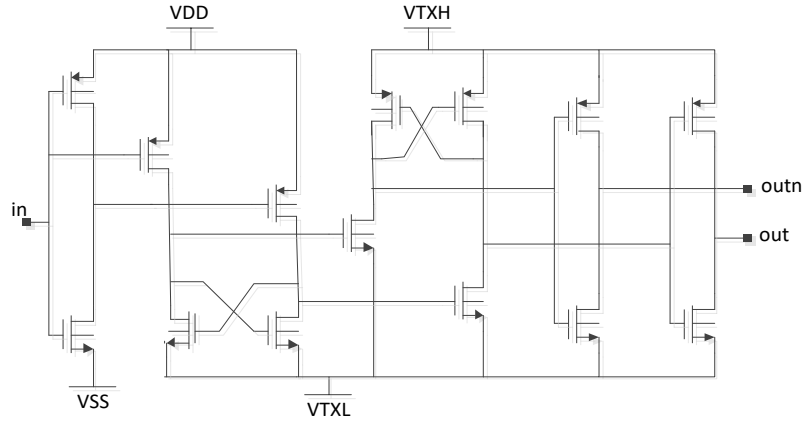
Fijando $L = 0.6$ por razones de comodidad para el futuro obtenemos el tamaño mínimo permitido por la tecnología, es decir $W=0.6$, $L=0.6$.

Para la etapa digital, partimos del tamaño mínimo $W = 0.5$, $L = 0.18$. Y partimos de los cálculos del primer inversor del buffer para mantener simetría de nuevo. De modo que el tamaño para el MOS tipo p, será (fijando el mismo tamaño de L) 1.5.

Para las etapas encargadas de convertir los niveles de tensión repetimos la misma operación, salvo que esta vez sobredimensionamos un 10% la etapa n con el objetivo de adquirir mayor velocidad en las caídas.

5.2.2.3 Level shifter

Partiendo del diseño anterior añadimos una etapa intermedia homóloga para que se ocupe de ajustar el nivel bajo a un nuevo nivel de tensión, podemos ver el resultado en la siguiente imagen.



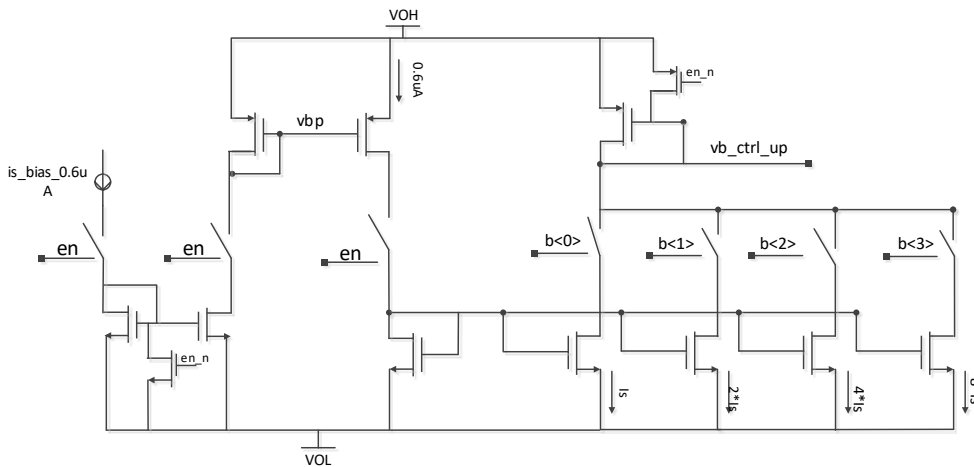
47. Level shifter con nivel alto y bajo programable.

5.2.3 Biasing

Finalmente diseñaremos la estructura encargada de programar las tensiones de vb con la que se controlará el slew. La estructura de biasing estará conformada de una serie de espejos de corriente que nos permitirá dar fracciones de la corriente del bandgap desde 1 hasta 16 veces, conformando así las distintas tensiones de vb.

Para el diseño de esta estructura, se ha realizado el diseño de un espejo de corriente, que debería ser capaz de espejar $I_{dc}/16$, y el resto de transistores serán el mismo transistor con distinta multiplicidad, consiguiendo así, que pueda ser elegida la corriente que se pretende espejar. Para programar la corriente se usa una serie de interruptores mos con el tamaño mínimo que soporta la corriente que va a pasar, en algunos caso se ha sobredimensionado para que fueran capaces de soportar una corriente de 0.6572 mA. Por último el MOS MMO tiene las mismas dimensiones que el de control de slew homólogo, pues es donde se debe de producir la copia de corriente.

Los circuitos de Bias, quedan como se muestra en las figuras:



48. Esquemático para el control de slew de subida.

6 DISEÑO DE BÚFERES DE SEÑALES DE CONTROL DE UN PIXEL PARA TOF: IMPLEMENTACIÓN

When you love something, you just do not want to stop talking about it.

- Neil Gaiman -

6.1 Buffers

Una vez los cálculos han sido llevados a cabo debemos realizar una serie de pruebas para corroborar la bondad de nuestros datos. Aunque el esquema básico funciona sobre papel, debemos buscar realizar el esquema de forma funcional. Esto es añadir todos los componentes necesarios en un sensor real, como son interruptores que nos permitan activar o desactivar nuestros buffers, un modo de función rápida e interruptores de desacoplo.

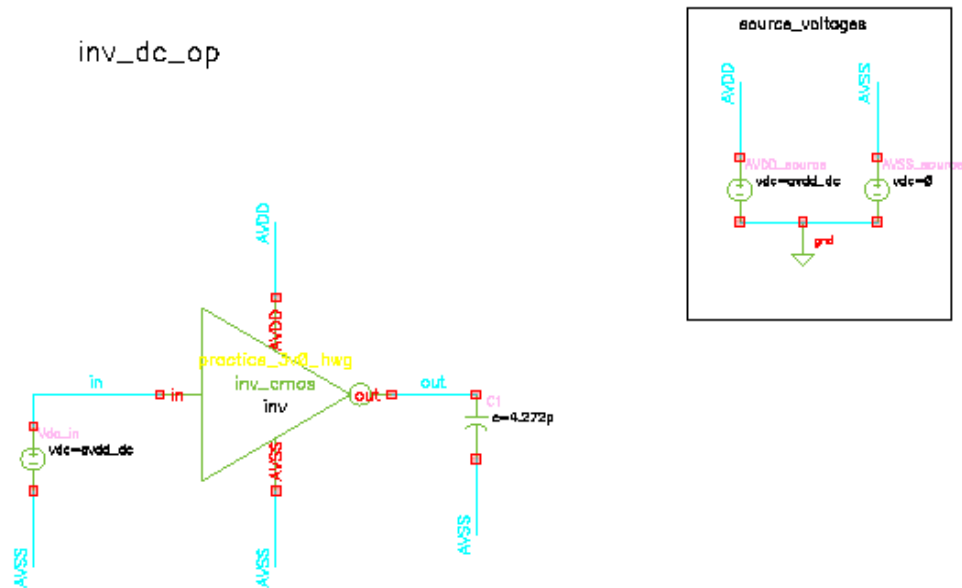
Con estas modificaciones buscamos realizar un bloque completamente funcional y que con pocas modificaciones puede ser reutilizable para futuros diseños. De esta forma somos capaces de realizar el diseño de estos bloques para otros sensores de carácter similar en menos tiempo y con mayor eficacia.

Realizaremos estas modificaciones para el buffer con control de slew, que es el objetivo final de nuestro diseño, sin embargo, someteremos a distintos test a todos los dispositivos, de la manera que se haría en el flujo normal del trabajo de diseño.

6.1.1 Buffer sin control de slew

Para la implementación del diseño se deben realizar distintos tipos de pruebas que corroborarán la bondad de

los cálculos realizados. Un buen comienzo es realizar unas pruebas en régimen estacionario donde la podamos apreciar la carga del pixel en el diseño calculado. Para ello, el primer test será una prueba del inversor que compone nuestro buffer sin control de ‘slew’, una vez la bondad del diseño sea corroborada, el diseño estará preparado para el siguiente paso.



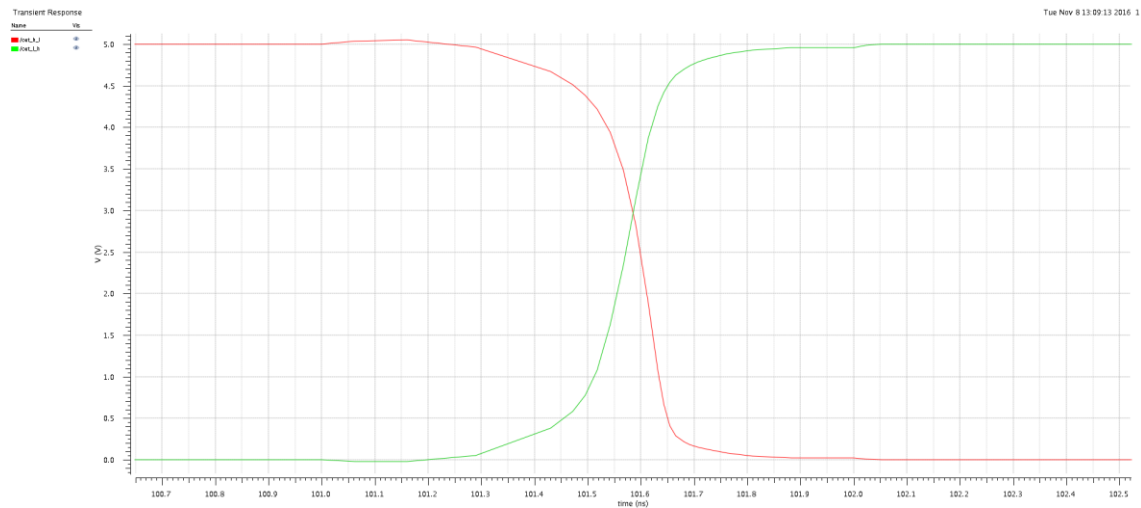
49. Esquema del ensayo DC del inversor con carga capacitiva.

La prueba consiste en utilizar una entrada de 5V y comprobar que la salida es efectivamente 0V, posteriormente se realizará la prueba inversa. Los resultados son satisfactorios, aunque sus resultados no revelan demasiado aparte del buen funcionamiento, el resultado de estas pruebas se añadirá en un anexo al final del documento.

La siguiente prueba será comprobar la simetría entre los tiempos de subida en el inversor, para ello someteremos al inversor a una prueba transitoria donde se varía el valor de la tensión de entrada de un nivel alto a un nivel bajo y posteriormente al revés, al superponer las dos transiciones, podremos observar donde se produce el corte. Nuestra intención ha sido que el corte se produzca en 2.5V. Es importante que las pruebas no se realicen una detrás de otra en el mismo plano temporal, de hacerlo así, puede verse falseado el experimento. El tiempo de bajada o de subida no será exactamente el mismo si partimos desde funcionamiento que si partimos desde apagado.

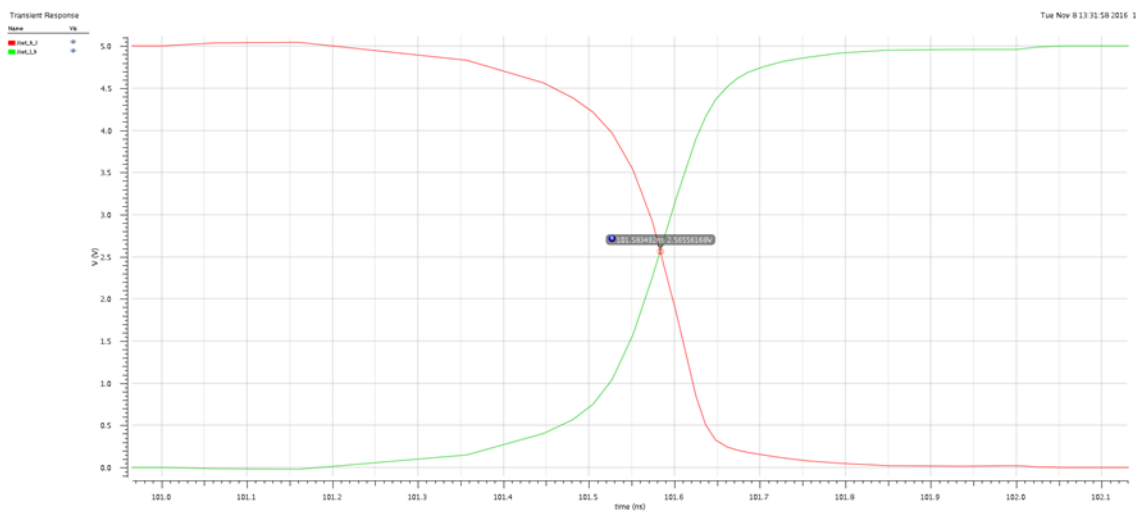
El punto de corte puede verse desvirtuado por lo anteriormente comentado hacia arriba, gracias al “desplazamiento” sufrido. Una buena forma de evitar este problema, consta en instanciar dos veces el bloque, y excitarlos de manera complementaria.

En la siguiente imagen podemos ver la comparación entre las transiciones.



50. Prueba de simetría del inversor.

Gracias a esta prueba podemos comprobar que no es perfectamente simétrico, pues el punto de corte se encuentra en 3.4V. El punto de corte se encuentra un poco alto, pues idealmente sería en 2.5V, sin embargo, es un error subsanable cambiando la relación de tamaños entre los transistores tipo p y tipo n. Tras unos cambios y varias pruebas de verificación conseguimos un resultado más apropiado, que se muestra en la siguiente imagen.

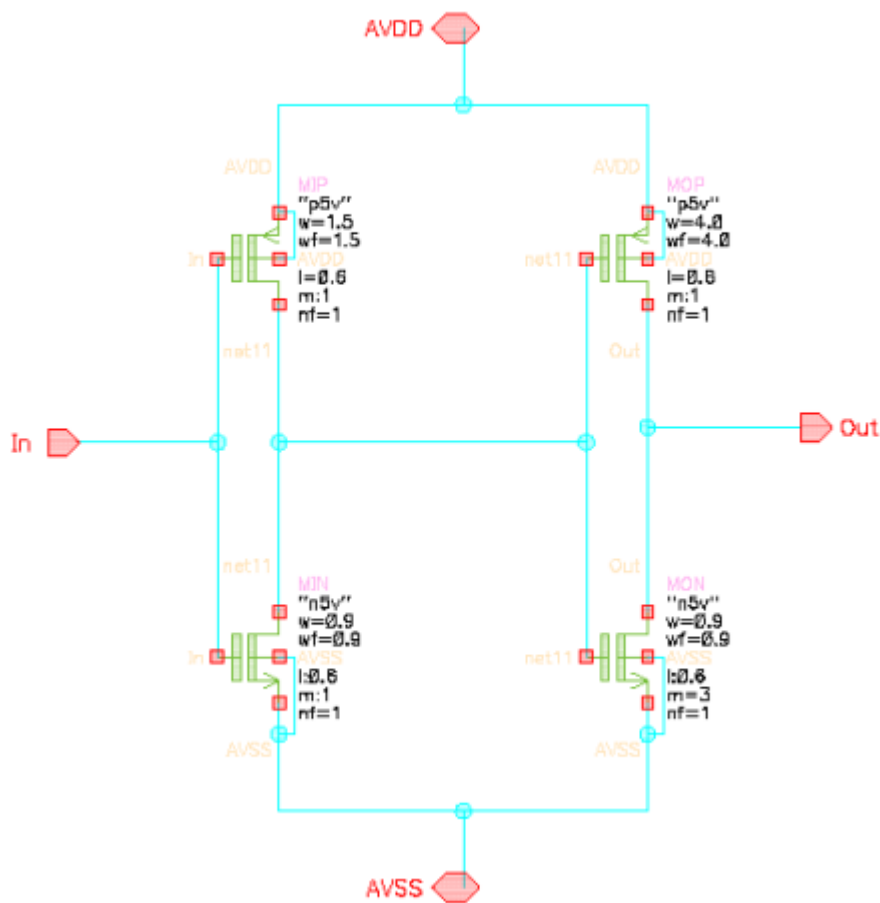


51. Comparación entre transiciones, punto de corte corregido.

Una vez comprobada la bondad de nuestro diseño procedemos a incluir la pre-etapa que anteriormente hemos calculado, recordamos, gracias a la fórmula:

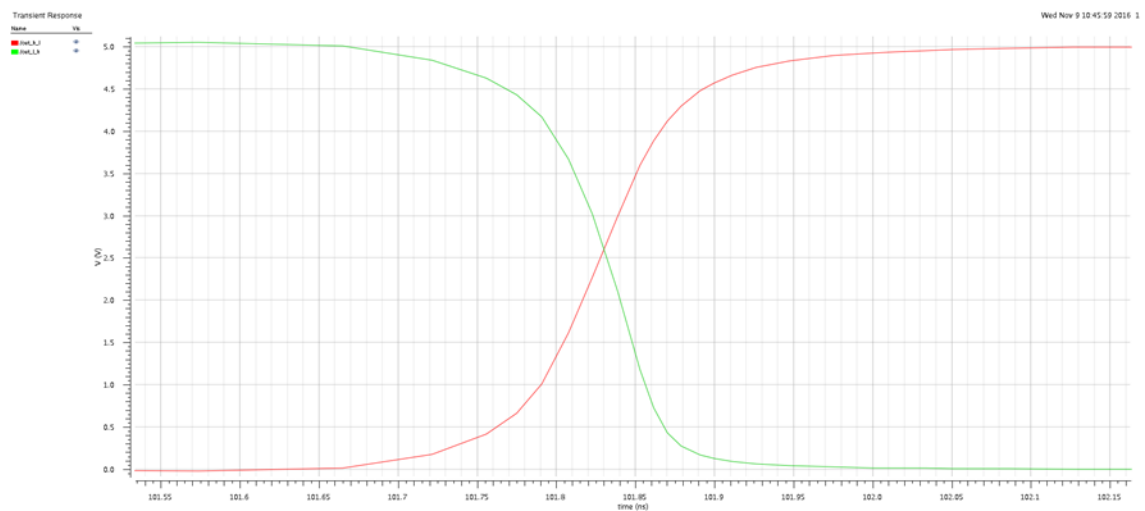
$$Me^n$$

El diseño queda como se muestra en la siguiente imagen:



52. Esquema del buffer con pre-etapa.

Volvemos a corroborar la simetría del buffer, esta vez, con la pre-etapa implementada.



53. Comprobación de tiempos del buffer.

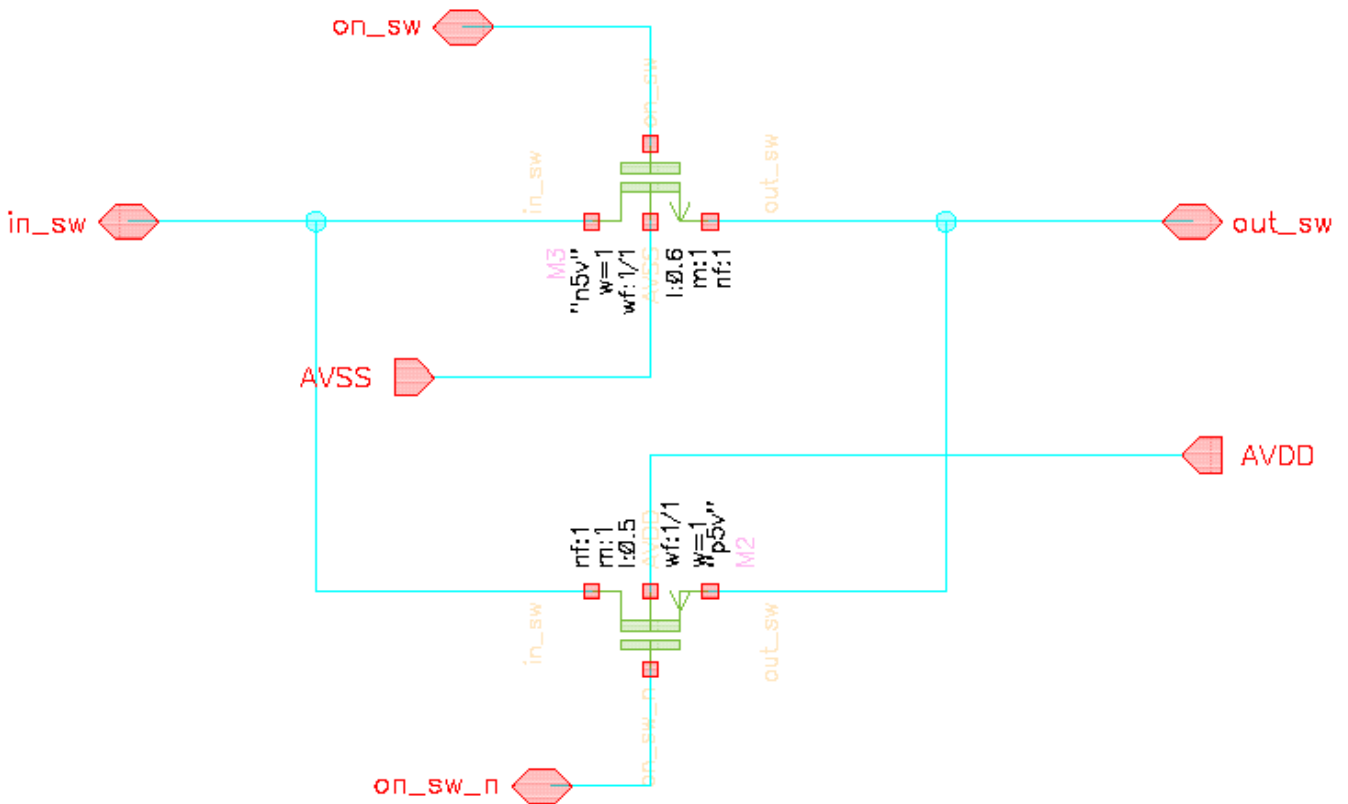
6.1.2 Buffer con control de slew

Para poder implementar el buffer con control de slew, debemos asegurarnos de hacer un diseño robusto. No se puede perder de vista que este bloque estará rodeado de otras arquitecturas y además una buena práctica es implementar flexibilidad en la actuación de nuestro diseño.

Por un lado, es interesante implementar el apagado del bloque. De este modo nos aseguramos que el dispositivo se apagará, al implementar un cierre a este nivel de arquitectura, la fila de píxeles no se verá afectada por las operaciones de otros bloques. El apagado en este bloque es de gran relevancia.

Por otro lado, la desconexión de este bloque con los demás también ha de tenerse en cuenta, evitando problemas de conducción espontánea (o en general de interacción entre bloques cuando no se desea) aumentando además la reusabilidad del bloque.

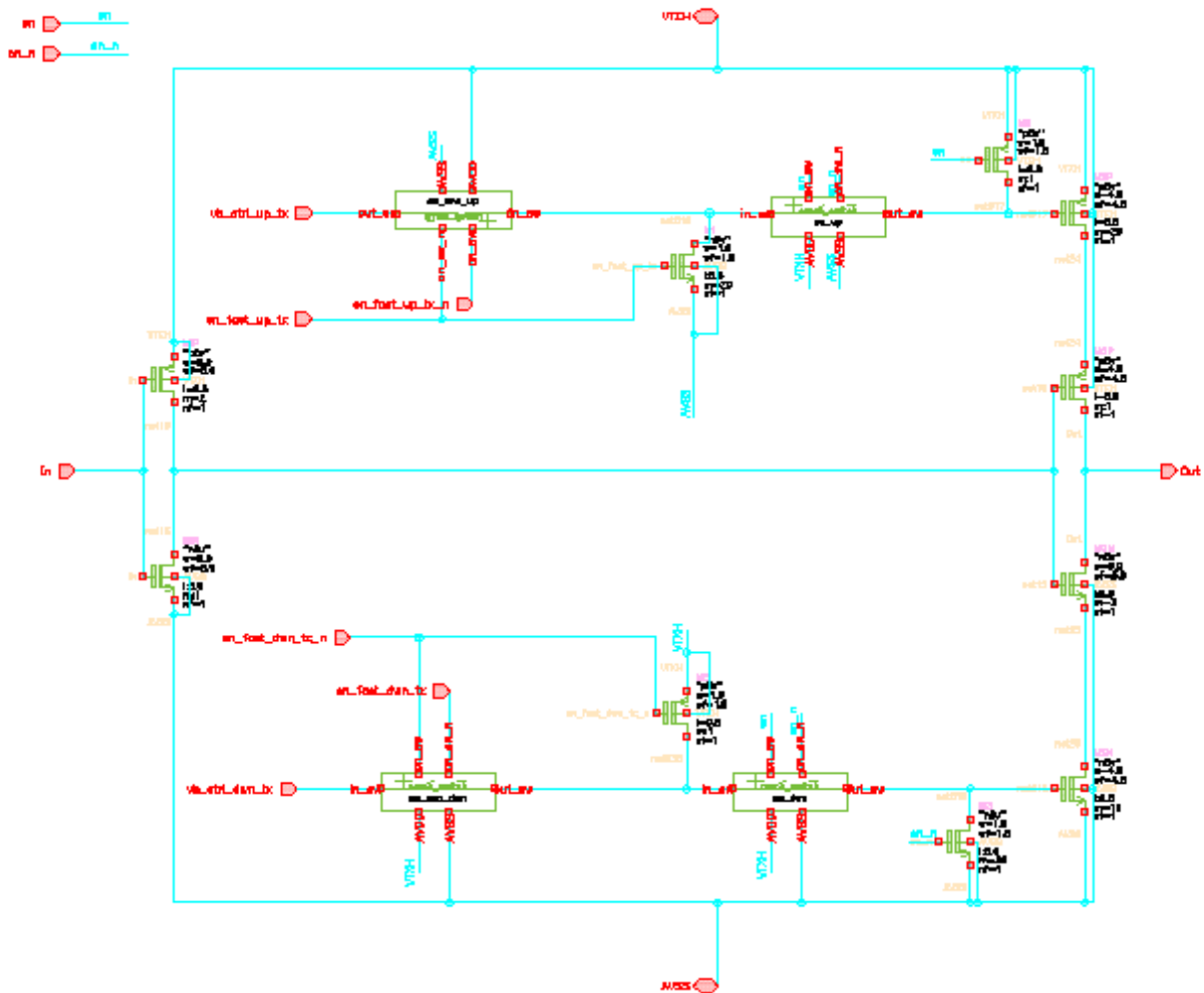
Por ello se presenta el siguiente diseño, en el que no profundizaremos.



54. Interruptor CMOS.

En la ilustración anterior se muestra un interruptor CMOS con este diseño podemos asegurarnos de cortar la corriente en la línea $in_sw - out_sw$ con rapidez. Haciendo uso de este interruptor alcanzaremos las metas de corte local y de separación de bloques individuales.

Con esto en mente, el diseño queda como se muestra en la siguiente ilustración.



55. Esquema de buffer con control de slew.

Gracias a los interruptores, podremos apagar la salida del bloque, o bien apagar el biasing del mismo, permitiendo que se aplique la velocidad máxima cuando sea requerido.

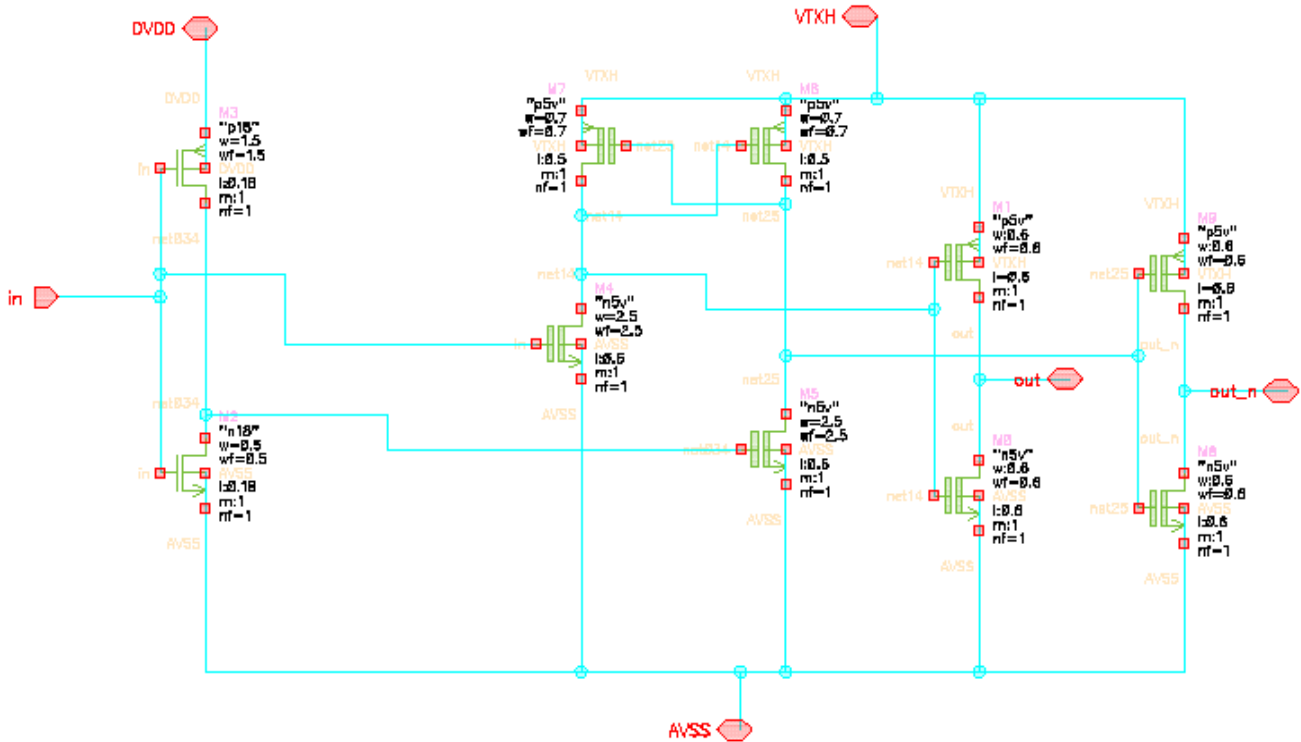
De este modo tendremos nuevas entradas y salidas en el bloque, correspondientes a la alimentación de los interruptores, en_fast encargada de aplicar la velocidad máxima posible y las señales complementarias de las mismas. Como añadido, se han introducido varios MOS encargados (juntos a los interruptores) del corte, o bien de la aplicación de tensión necesaria para aplicar velocidad máxima.

Si nos detenemos un poco para analizar el bloque, podemos ver que el primero de los interruptores se encargará de conectar o desconectar el biasing, esto es interesante no solo por la reusabilidad sino porque para poder aplicar corte o tensión máxima es más seguro que la línea responda solo a una conexión, evitando que puedan ocurrirse comportamientos no deseados por caídas de tensión en la línea o derivados.

En cuanto al segundo interruptor, desconectará la línea desde el MOS encargado de aplicar tensión máxima del MOS encargado del corte, de nuevo es una medida de seguridad para asegurar que el comportamiento final del bloque es el deseado.

6.2 Level shifter

En este subapartado vamos a implementar los distintos level shifters en el orden que se han presentado en el capítulo anterior con los tamaños calculados anteriormente. El primer esquema presentado es el mostrado en la siguiente ilustración.

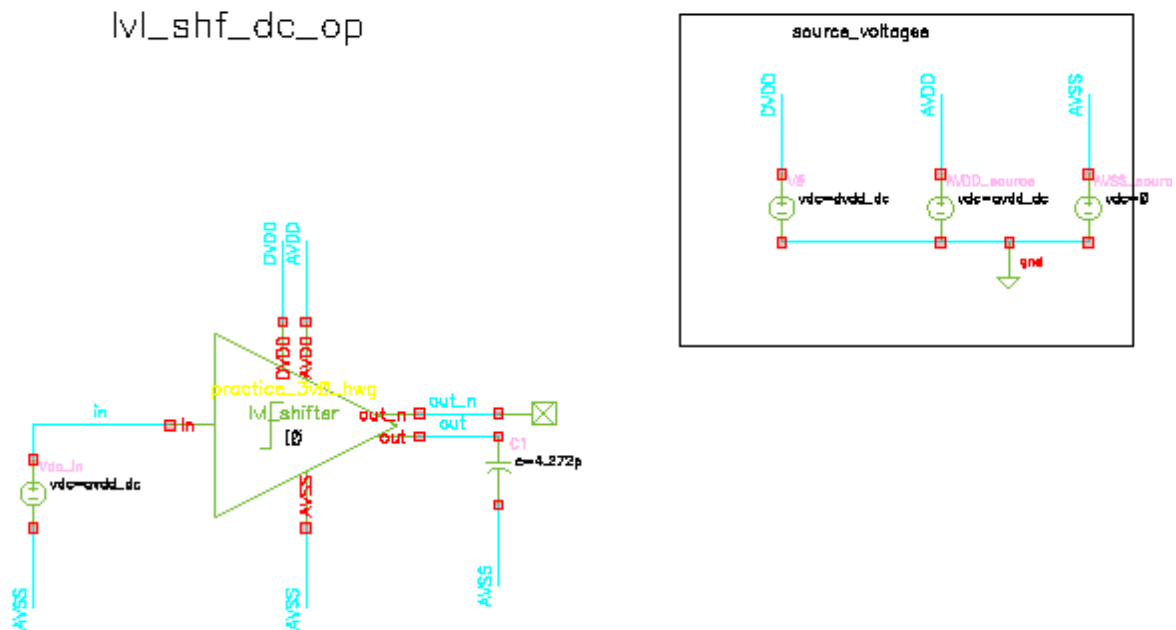


56. Esquema 'Level shifter' básico.

Los niveles de tensión de las partes digital y analógica se corresponden a a 1'8 V y 5V respectivamente. Sin embargo la masa de ambas partes es compartida. Siguiendo el mismo procedimiento para todos los bloques primero realizmos una prueba del bloque de forma aislada, para simular la carga a la salida del bloque, incluiremos un condensador de carga similar a la capacidad parásita de la línea, de este modo podremos corroborar la bondad del dispositivo asegurándonos que cumple con las expectativas propias a su diseño.

La primera prueba que realizaremos será una simulación en DC, comprobaremos que para una entrada de 1'8V obtenemos una salida de 5V, y que para una entrada de 0V obtenemos una salida igual. Con esta simulación nos aseguramos que el dispositivo cumple su función.

Recordamos que este bloque irá conectado al bloque del buffer, su única función será adaptar el dominio digital al dominio analógico, para que más tarde el bloque del Buffer se encargue de repartir la señal con una corriente adecuada a la carga exterior.



57. Layout de la prueba DC del primer level shifter.

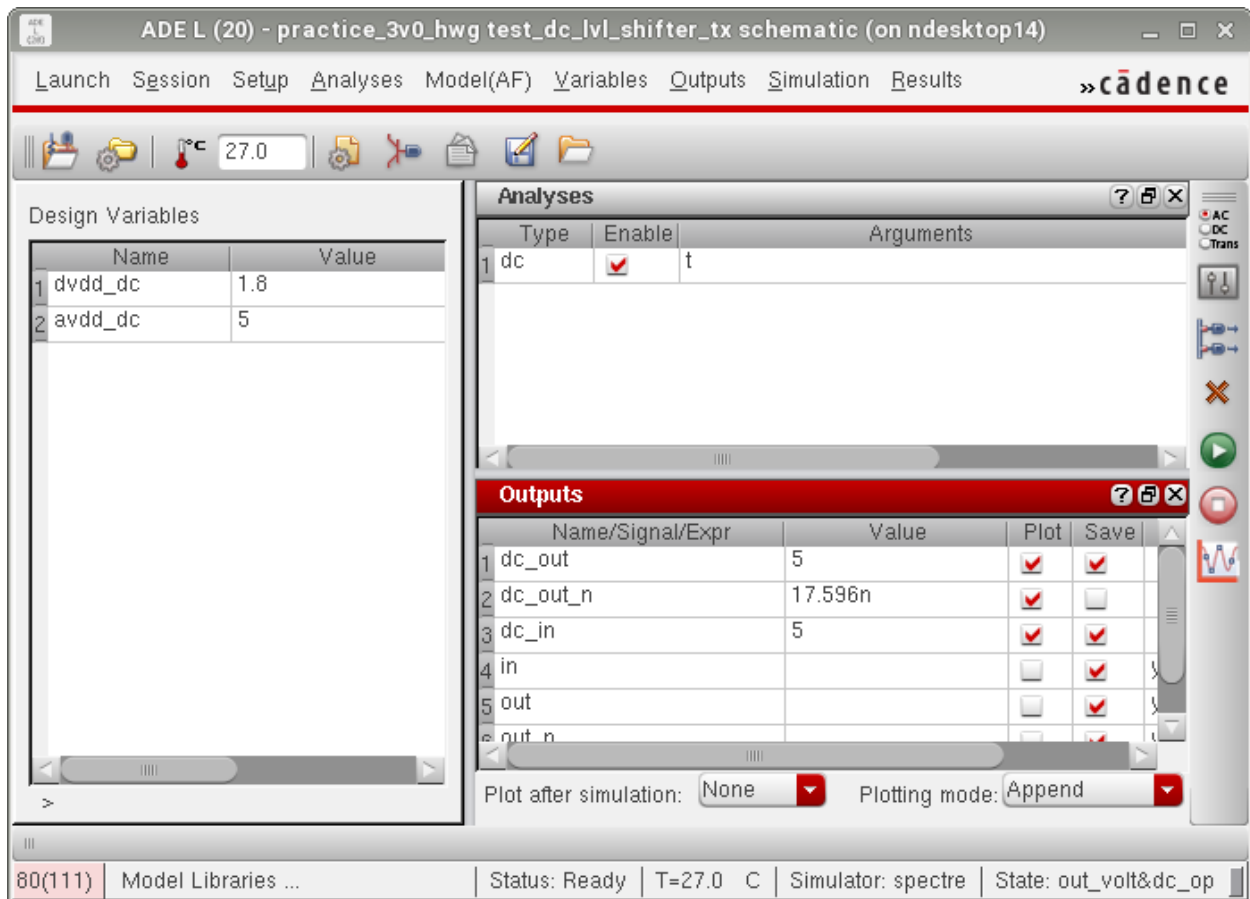
Dado que el bloque no funcionará en estático se hace imperativo comprobar que el diseño funcionará con una tensión variante, pues debemos recordar que las señales que atravesaran todo el diseño variarán continuamente. Con el mismo esquema pero una entrada tren de pulso podemos comprobar si, efectivamente, el comportamiento es el esperado.

6.2.1 Level shifter con niveles de salida programable

El cambio entre dominios es un lugar crítico debido a muchos factores, uno de ellos es la masa, si bien con una masa único todo es más simple, cuando tendemos a separar los dominios totalmente, conseguimos numerosas ventajas, una de ellas es la limpieza del dominio digital.

Como ya se había comentado, al compartir una misma tierra, los cambios constantes en el dominio analógico (siendo además estos a tensiones relativamente altas para el dominio digital) afectan empeorando las señales del dominio digital, empeorando el rendimiento.

Los resultados no se ven apenas afectados, siendo el punto más afectado el área utilizada y la complejidad de la implementación física. En la siguiente imagen podemos comprobar el similar resultado.



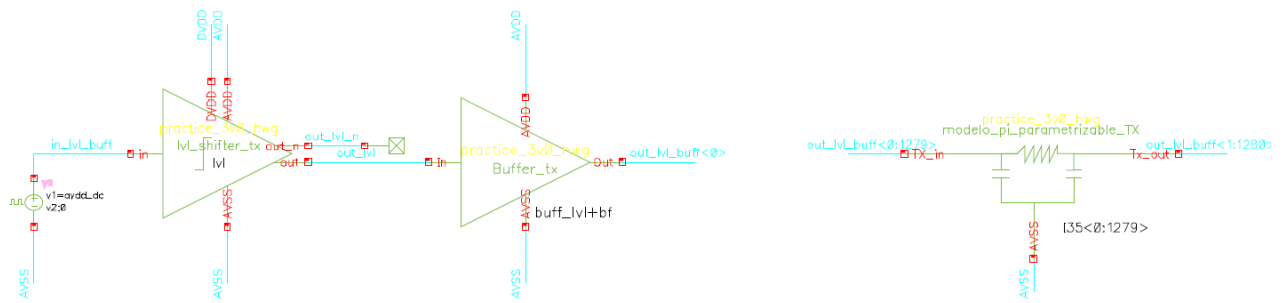
58. Pruebas dc_OP Levle shifetr niveles programables.

Al hablar de complejidad en la implementación física, nos referimos a la dificultad para implementar lo que a vistar parece un diseño muy parecido. Al separar las masas, debemos hacer una separación física entre los pozos, y además debemos reservar un buen espacio vacío entre los pozos, de manera que no se afecten entre ellos.



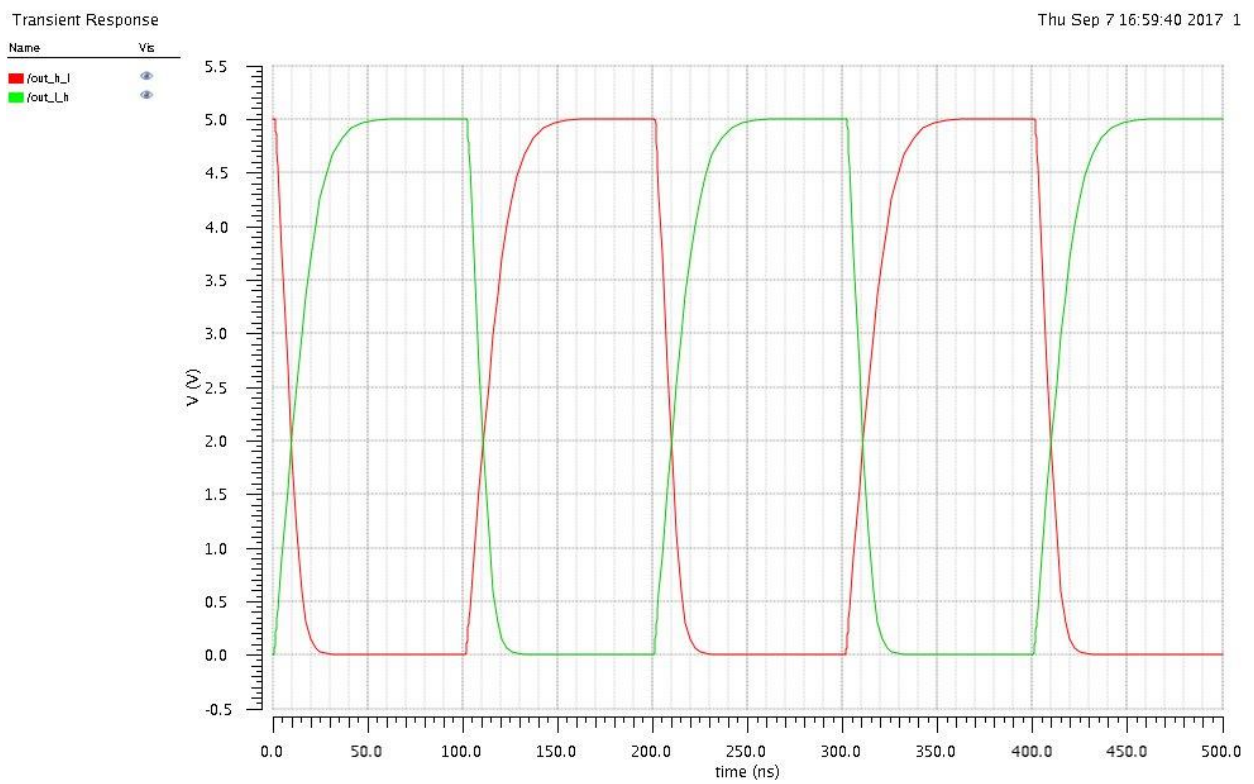
59. Layout Level Shifter con tierras separadas.

A continuación se muestra como la implementación de ambas topologías afectan al resultado final. En esta prueba lanzamos un test transitorio con el level shifter y el buffer conectados. En esta imagen podemos ver el esquemático de la prueba.

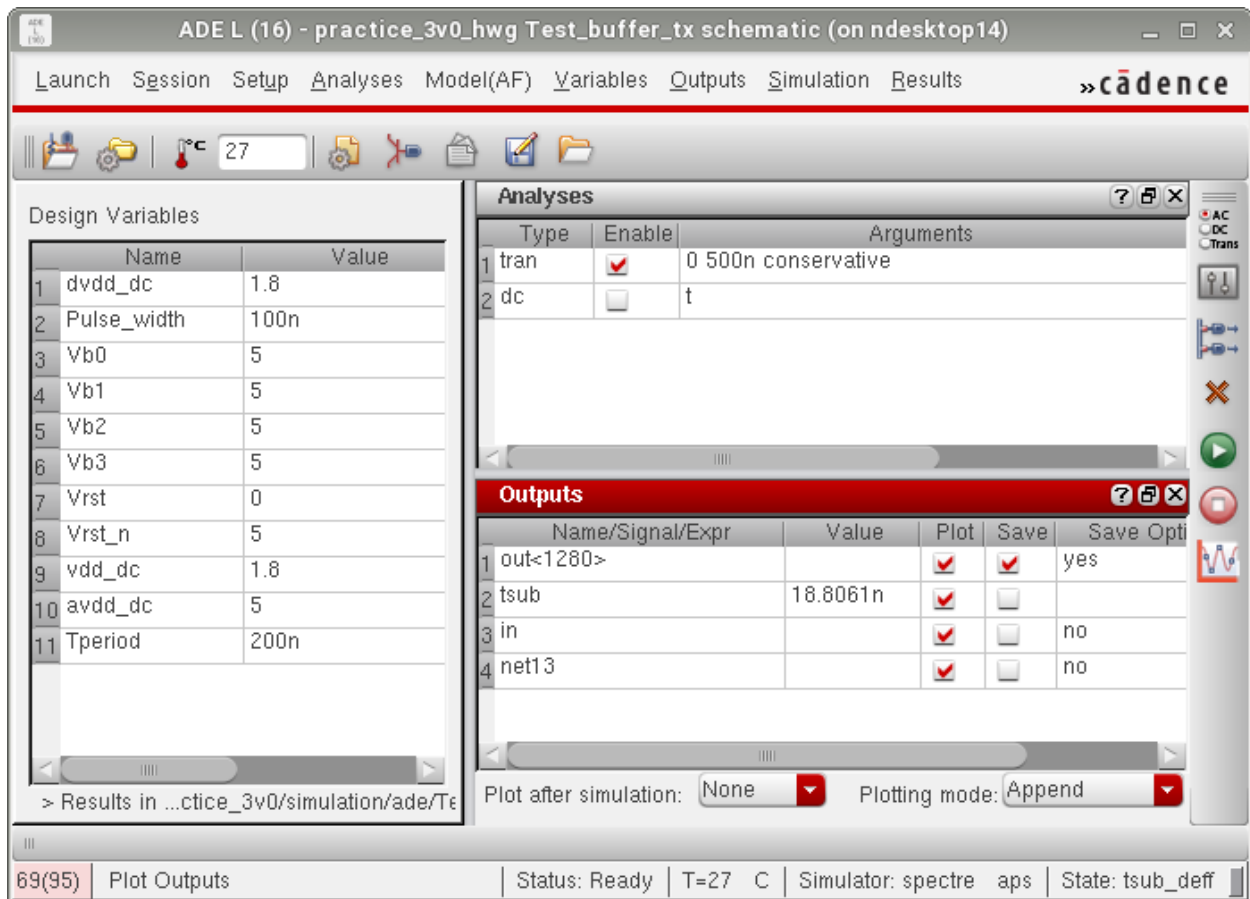


60. Prueba transitoria level shifter más buffer

Los resultados de esta prueba son los siguientes.



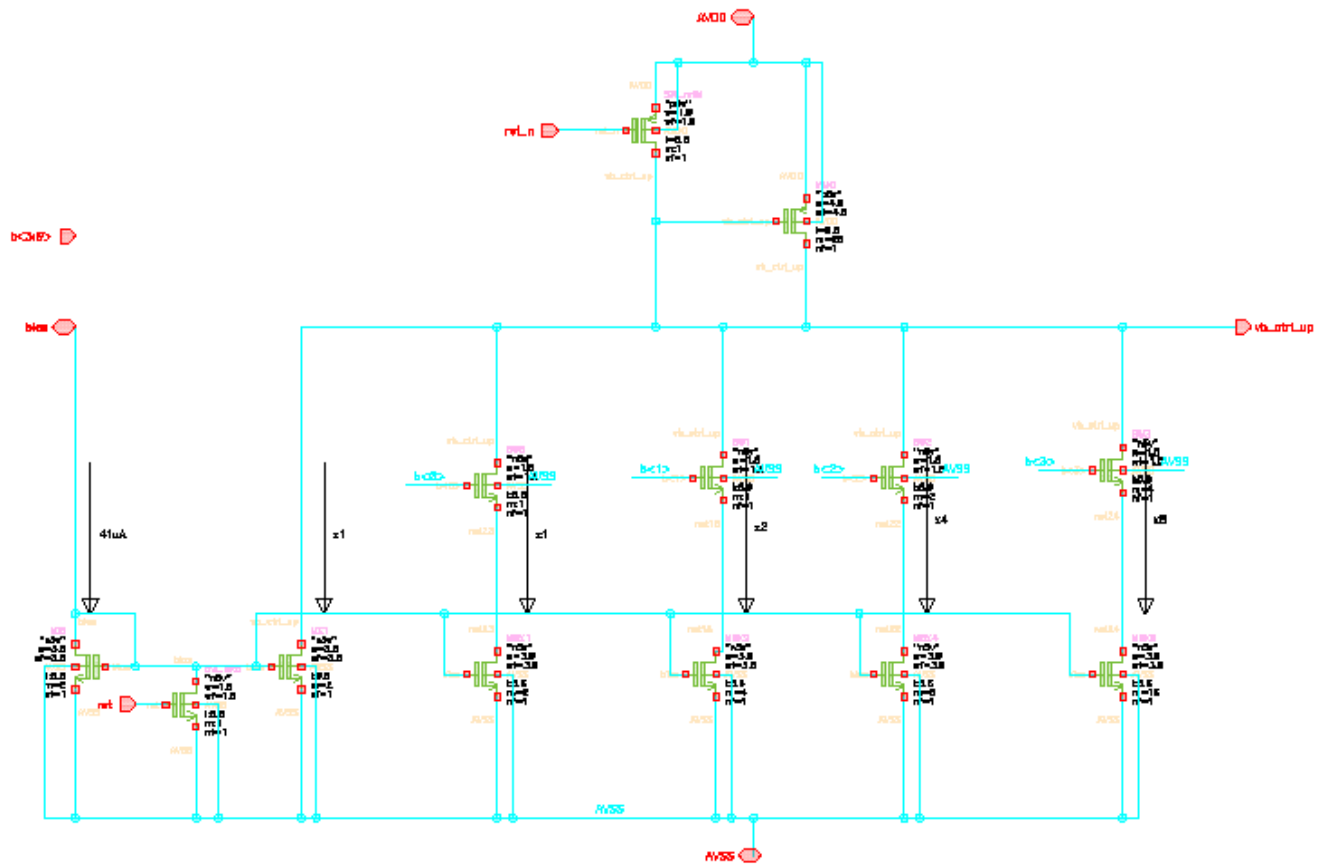
61. Pruebas transitorias level shifter+buffer



62. Tiempo de subida pruebas buffer + level shifter.

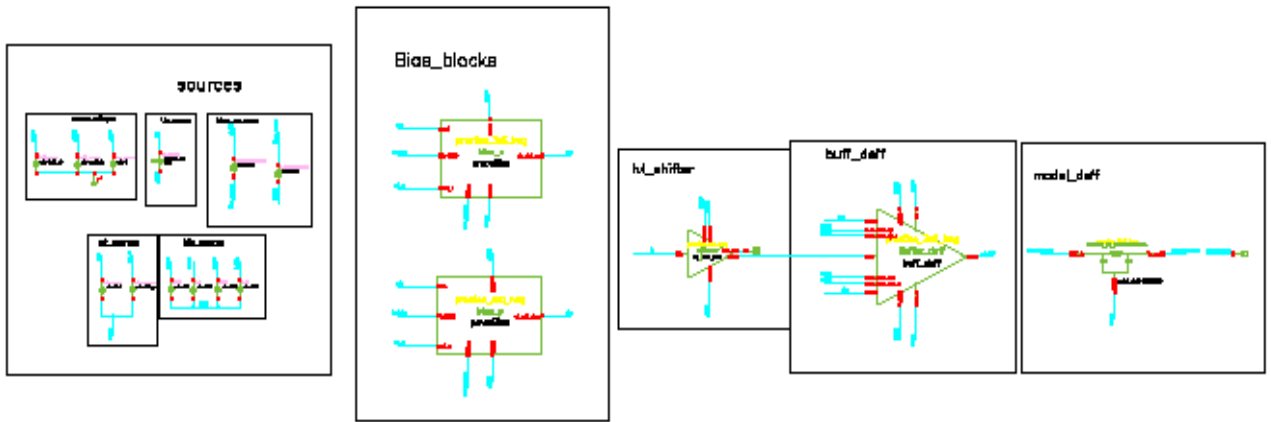
6.3 Biasing

Atendiendo a los cálculos realizados en el apartado anterior, los bloques de Biasing, encargados de proporcionar la tensión necesaria para controlar el slew de los buffers, quedan como se muestra en las siguientes figuras.



63. Esquema de Biasing para el control de la pendiente de subida.

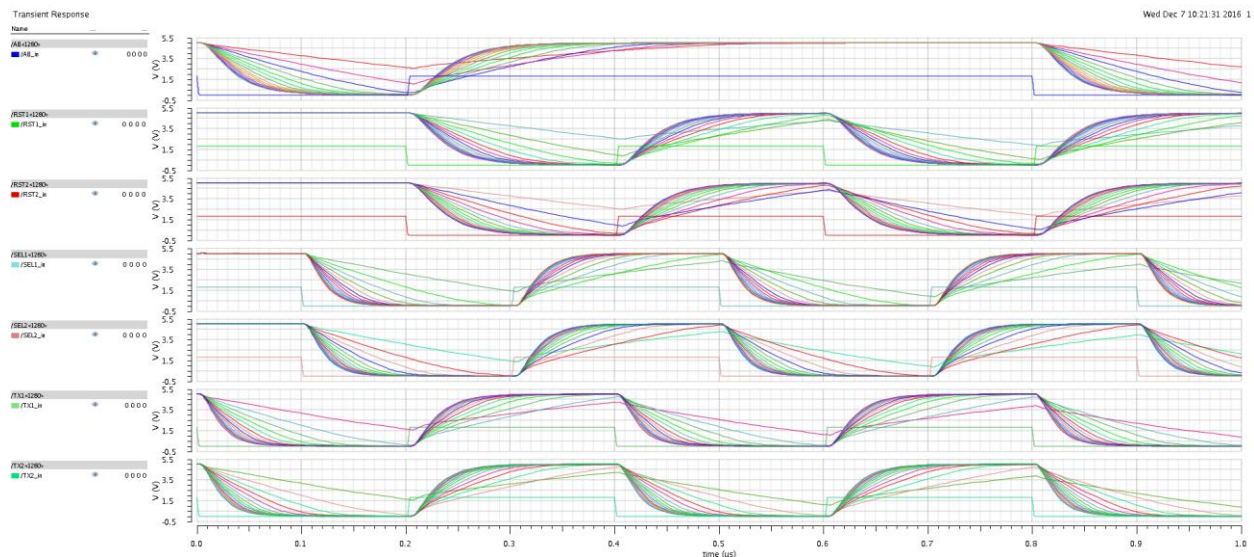
Tsub_all_blocks



65. Simulación de todos los bloques con modelado de salida.

Incluimos en la simulación un bloque que se encargará de proporcionar las corrientes necesarias para que podamos simular distintos puntos de operación del Biasing. De este modo cubrimos con todas las posibilidades de funcionamiento de la fila.

Corremos varias simulaciones en las que cubrimos todas las posibilidades mencionadas. Esto además lo realizamos para todas las señales dadas. El resultado es el siguiente.



66. Simulación de todas las pendientes para todas las señales usando todos los bloques.

Las medidas de los resultados se recogen en la siguiente tabla.

| <i>Point</i> | <i>TS_AB</i> | <i>TF_AB</i> | <i>TS_SEL1</i> | <i>TF_SEL1</i> | <i>TS_RST1</i> | <i>TF_RST1</i> | <i>TS_TX1</i> | <i>TF_TX1</i> |
|--------------|--------------|--------------|----------------|----------------|----------------|----------------|---------------|---------------|
| 1 | 470.7n | 408.8n | 291.4n | 255.6n | 429.9n | 379.5n | 319.6n | 281.5n |
| 2 | 255.9n | 226n | 155.9n | 140.9n | 234n | 210.6n | 172.4n | 155.3n |
| 3 | 181n | 160.7n | 110.7n | 99.75n | 166n | 151.2n | 122.3n | 109.2n |
| 4 | 142.1n | 126.7n | 86.32n | 78.35n | 129.6n | 119.8n | 95.89n | 85.83n |
| 5 | 120n | 106.7n | 72.84n | 65.87n | 109.3n | 101.7n | 80.82n | 71.36n |
| 6 | 105.6n | 93.66n | 63.79n | 57.75n | 96.12n | 90n | 70.79n | 62.02n |
| 7 | 95.79n | 84.51n | 58.15n | 52.09n | 87.38n | 82.54n | 64.54n | 55.86n |
| 8 | 88.62n | 77.68n | 53.85n | 48.16n | 80.66n | 76.51n | 59.75n | 51.24n |
| 9 | 83.45n | 73.13n | 51.03n | 45.19n | 76.64n | 72.71n | 56.44n | 47.59n |
| 10 | 80.02n | 69.59n | 48.83n | 42.59n | 73.13n | 69.81n | 54.12n | 44.96n |
| 11 | 77.04n | 66.26n | 47.07n | 41.05n | 70.47n | 67.25n | 52.1n | 42.99n |
| 12 | 75.2n | 64.21n | 45.54n | 39.74n | 68.52n | 65.86n | 50.87n | 41.28n |
| 13 | 73.34n | 62.78n | 45.01n | 38.57n | 67.36n | 64.75n | 50.17n | 40.07n |
| 14 | 72.26n | 61.1n | 44.29n | 37.77n | 66.03n | 63.75n | 49.23n | 39.04n |
| 15 | 71.58n | 60.04n | 43.69n | 37.05n | 65.28n | 63.35n | 48.57n | 38.02n |
| 16 | 70.81n | 59.46n | 43.09n | 36.26n | 64.85n | 62.56n | 48.34n | 37.44n |

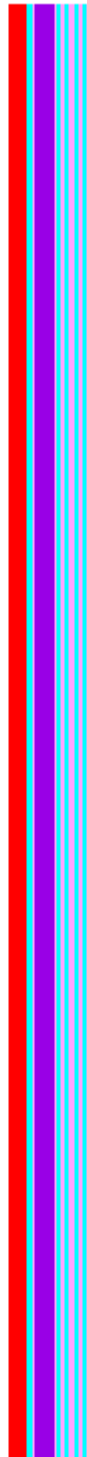
Tabla 3. Resultados de tiempo s de subida a y bajada de todas las señales.

Esta simulación se ha realizado en entorno nominal, lo que significa que se ha simulado para una temperatura de 27° y una tensión analógica de 5V.

6.4 Implementación física.

A lo largo de este subapartado exploraremos en detalles las implementaciones físicas individuales y de conjunto de todo el dispositivo. Comenzaremos con una vista general e iremos concretando pieza por pieza.

En primer lugar se presenta la implementación física de la matriz completa:



Para este diseño se ha buscado la menor área posible, utilizando las distintas líneas de distribución lo mejor posible para ahorrar espacio. La razón de buscar un área pequeña no es solo el tamaño en sí, sino que además mientras menor sean las pistas que deben atravesar las señales, menos se verán afectadas por estas.

Si hacemos un zoom en un slice podremos observar más detenidamente los detalles de la imagen.



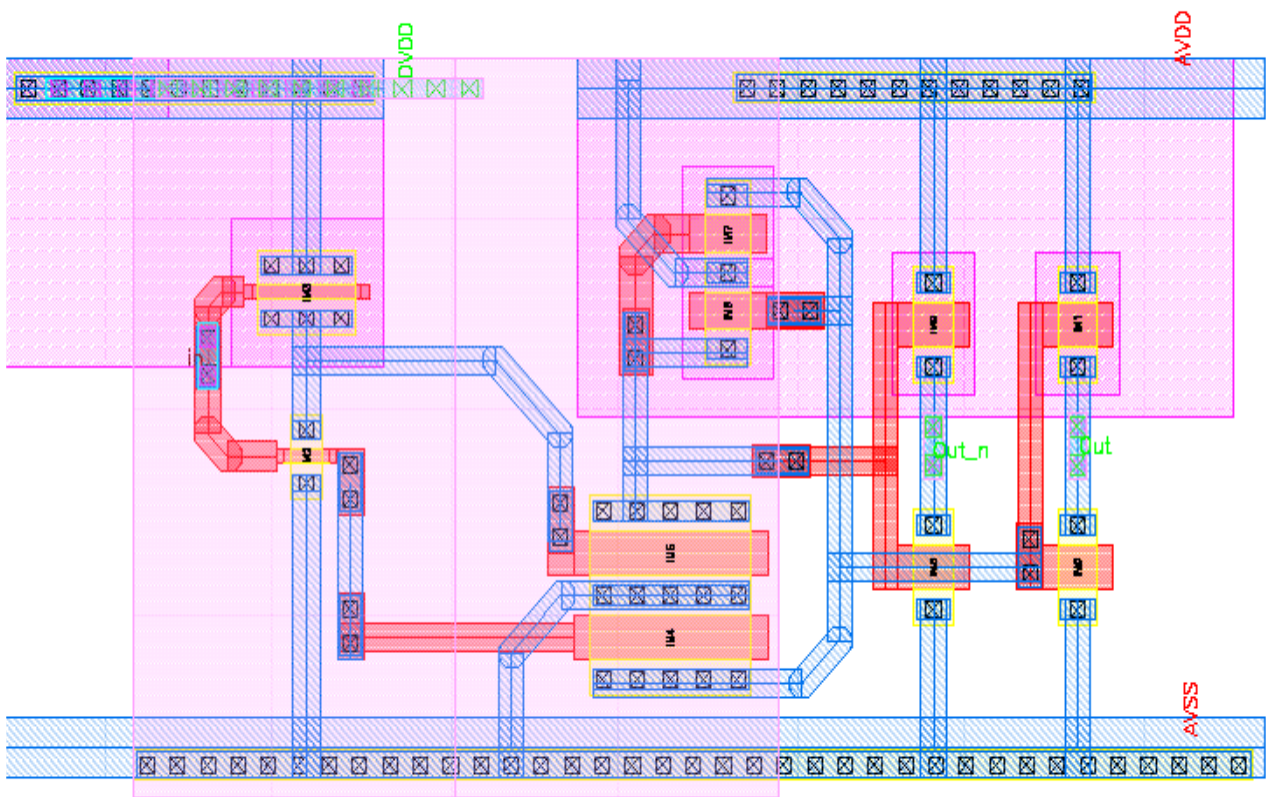
Como podemos observar en la zona rodeada de azul se encuentran los level shifters, encargados de cambiar de

dominio, en concreto, la zona rodeada de amarillo se corresponde con la parte dedicada al dominio digital de los mismos. Aquí es donde se encuentra la división de dominios entre digital y analógico.

La zona rodeada de color rojo corresponde a los buffers de las señales, encargadas tan solo de continuar la señal hasta la fila de pixeles.

6.4.1 Level shifter

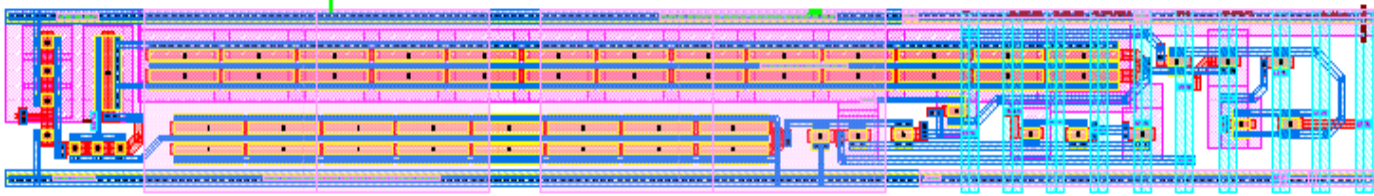
En la siguiente imagen se presenta un zoom al level shifter. Como se ha comentado antes, la zona rodeada de amarillo corresponde al dominio digital del dispositivo, aquí es donde separamos las tierras de ambos dominios.



Es importante separar ambos pozos con un anillo de guarda, minimizando así el impacto que puede tener una masa sobre la otra.

6.4.2 Buffers

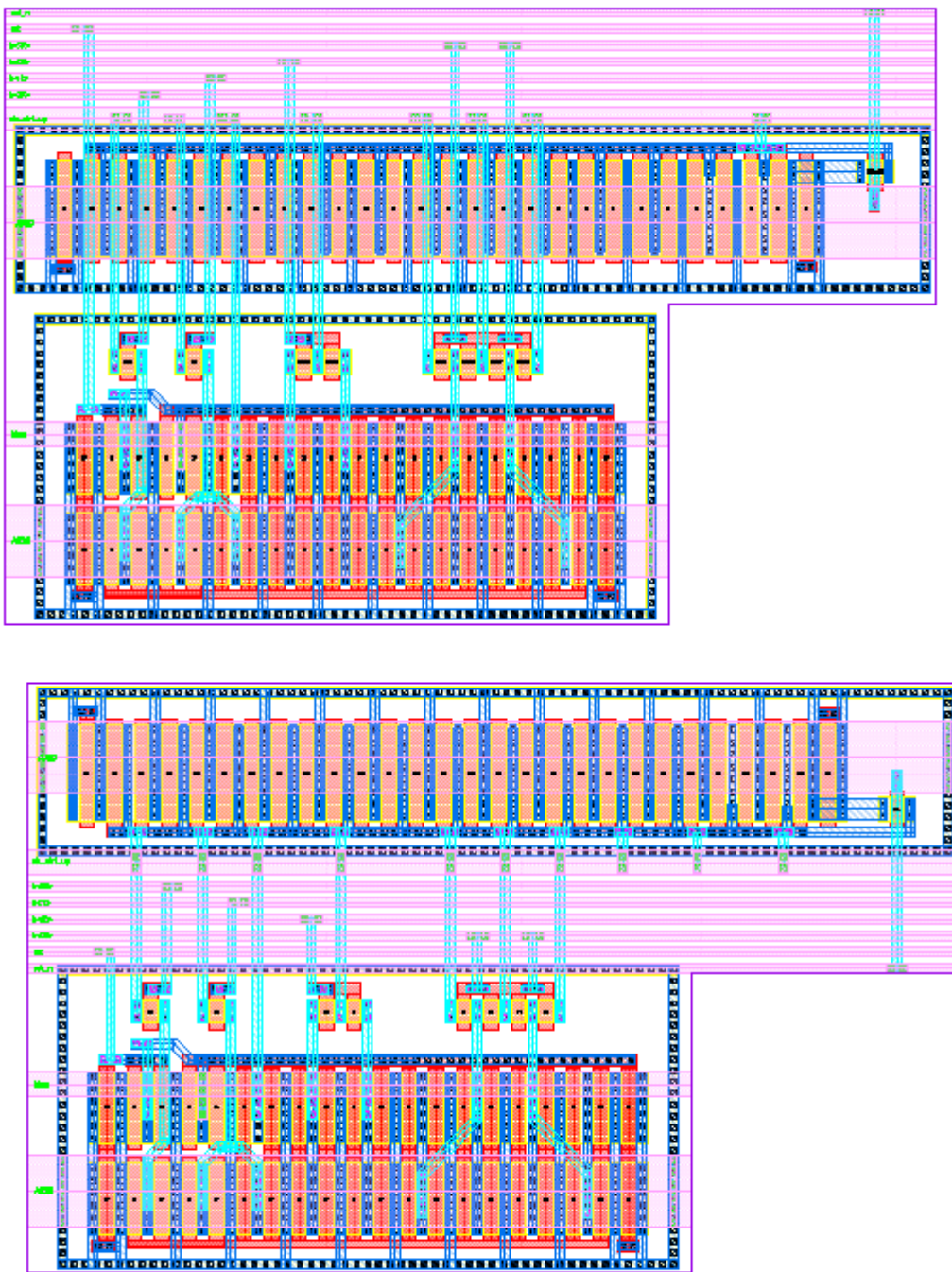
En la siguiente imagen se presenta el layout de los buffers



Se pretende en esta implementación conseguir una línea de señales lo más limpia posible, dado que los buffers están colocados de manera secuencial, es inevitable que constantemente las señales deban estar bajando y subiendo de nivel.

6.4.3 Biasing

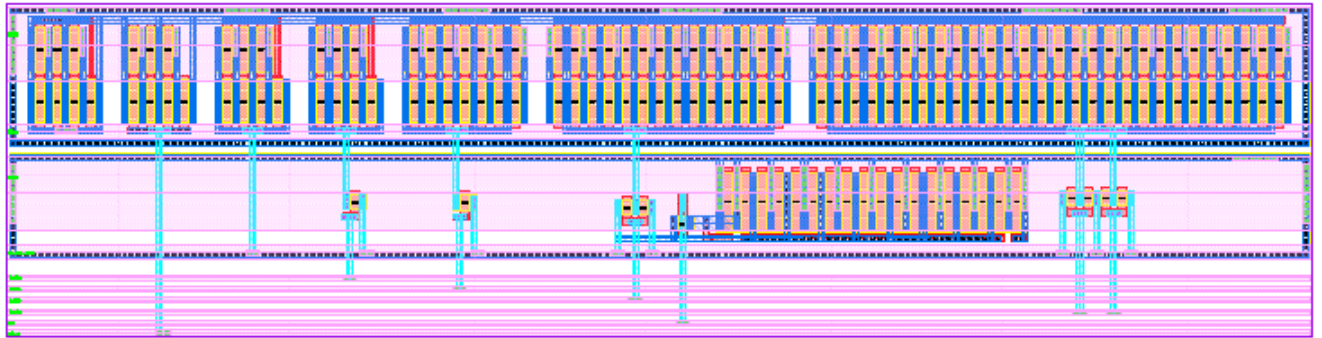
En las siguientes imágenes podemos observar la implementación de las dos estructuras encargadas del biasing. Por un lado el dispositivo encargado de controlar la pendiente de subida:



67. Layout del Biasing encargado del control de subida.

Podemos observar como se distribuyen los transistores para aprovechar el área al máximo, comparando esta disposición con los esquemas presentados en apartados anteriores, podemos ver como una fila de transistores se encargan de la copia de la corriente de entrada, mientras que el resto se ocupa de su multiplicación. Por último los últimos se encargan de controlar que transistores darán corriente y por tanto tensión a la salida, con esta implementación logramos separar aún más si cabe la parte de entrada del dispositivo de la zona de salida.

En la siguiente imagen presentamos la disposición física de la parte encargada de controlar la pendiente de bajada.



68. Layout del biasing encargado de la pendiente de bajada.

Aprovechamos además que estos dos dispositivos son muy parecidos para presentar distintas disposiciones posibles. Con esta disposición conseguimos una solución parecida en área pero que resulta más conveniente. Dado que la matriz de píxeles y en general todo el dispositivo tiende a ser más ancho que los dispositivos en biasing, debemos tener en cuenta a la hora de diseñar, que hay ciertas medidas insalvables.

De este modo, el ancho del control analógico de fila no puede verse más reducido con las disposiciones aquí mostradas, por lo tanto una buena práctica es alargar lo máximo posible el resto de periféricos para que en conjunto se ahorre espacio en área.

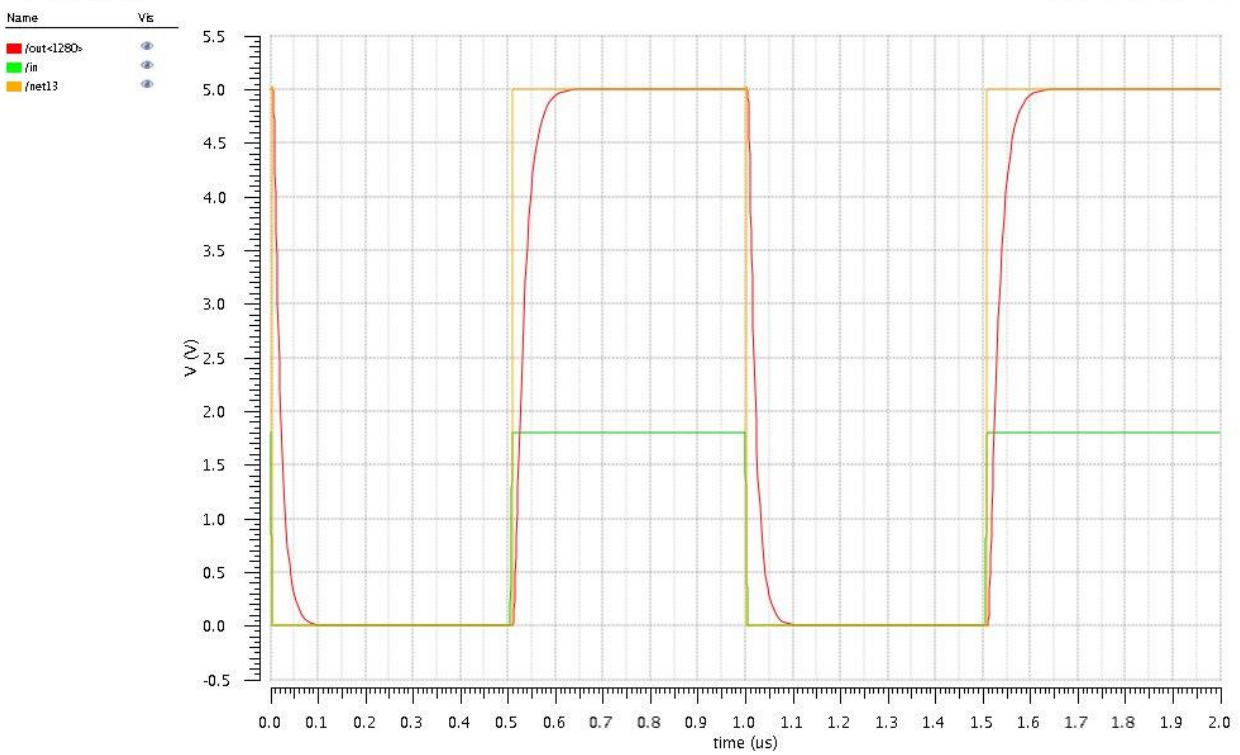
7 ANEXO.

Niño, o acabas la Carrera, o te vienes a la carpintería.

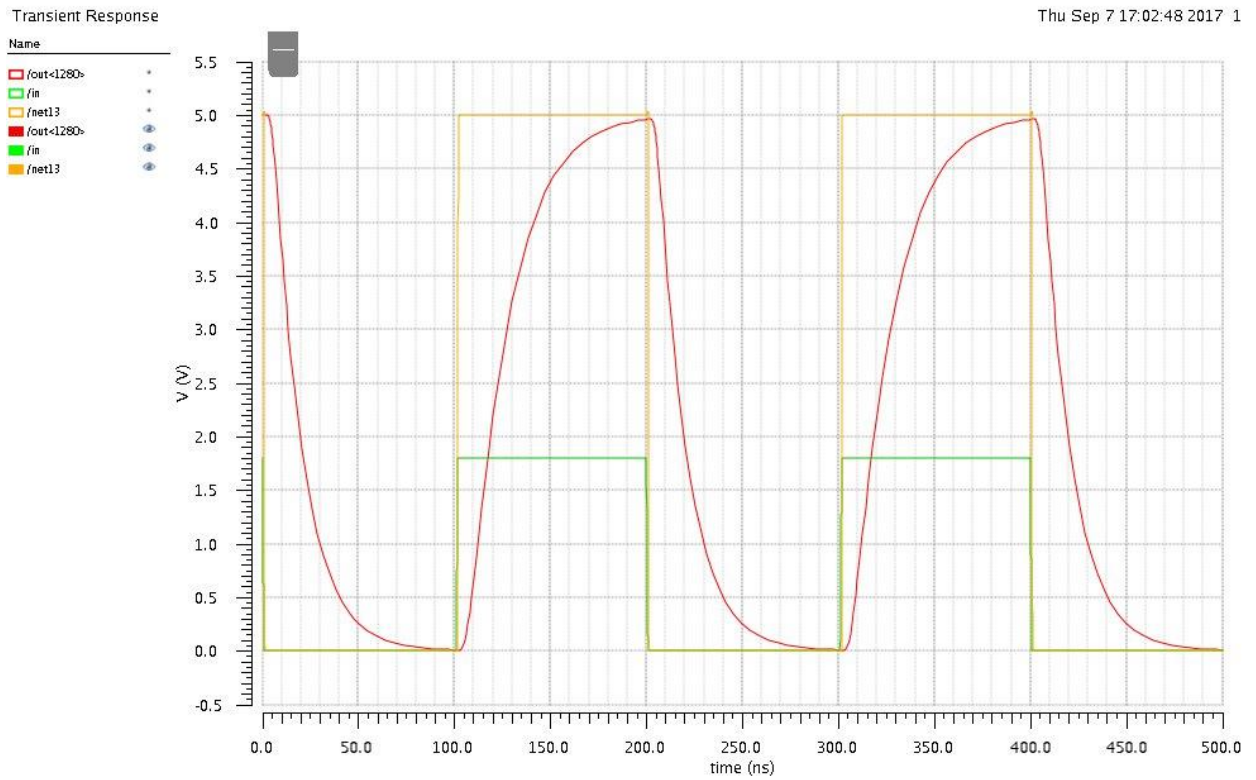
- Mi padre -

Transient Response

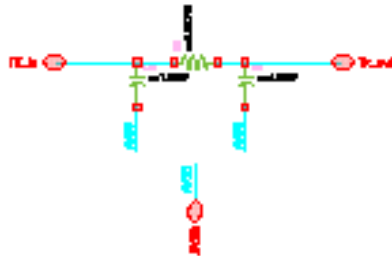
Thu Sep 7 17:01:40 2017 1



Anexo 1. Salida Buffer transitoria a máxima velocidad.



Anexo 2 . Salida transitoria del buffer a mínima velocidad.



Anexo 3. Modelo pi parametrizable.

ADE L (18) - practice_3v0_hwg test_dc_bias schematic (on ndesktop14)

Launch Session Setup Analyses Model(AF) Variables Outputs Simulation Results » cadence

27

Design Variables

| Name | Value |
|-----------|-------|
| 1 avdd_dc | 5 |
| 2 Vb0 | 5 |
| 3 Vb1 | 5 |
| 4 Vb2 | 5 |
| 5 Vb3 | 5 |
| 6 Vrst | 0 |
| 7 Vrst_n | 5 |

Analyses

| Type | Enable | Arguments |
|------|-------------------------------------|-----------|
| 1 dc | <input checked="" type="checkbox"/> | t |

Outputs

| Name/Signal/Expr | Value | Plot | Save | Save |
|-------------------------|-----------|-------------------------------------|-------------------------------------|------|
| 1 ldc_nbias | -1.01061m | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| 2 ldc_pbias | 952.834u | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| 3 block_bias_nmos/MMO/D | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | yes |
| 4 block_bias_pmos/M7/D | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | yes |

Plot after simulation: Plotting mode:

75(103) Load State ... | Status: Ready | T=27 C | Simulator: spectre | State: allcurrent&resets_bias

Anexo 4. Test DC biasing

The screenshot displays the Cadence ADE L (16) interface for a simulation. The window title is "ADE L (16) - practice_3v0_hwg Test_buffer_tx schematic (on ndesktop14)". The interface includes a menu bar (Launch, Session, Setup, Analyses, Model(AF), Variables, Outputs, Simulation, Results) and a toolbar with icons for file operations and simulation control. The temperature is set to 27°C.

Design Variables

| Name | Value |
|---------------|-------|
| 1 dvdd_dc | 1.8 |
| 2 Pulse_width | 100n |
| 3 Vb0 | 5 |
| 4 Vb1 | 5 |
| 5 Vb2 | 5 |
| 6 Vb3 | 5 |
| 7 Vrst | 0 |
| 8 Vrst_n | 5 |
| 9 vdd_dc | 1.8 |
| 10 avdd_dc | 5 |
| 11 Tperiod | 200n |

Analyses

| Type | Enable | Arguments |
|--------|-------------------------------------|---------------------|
| 1 tran | <input checked="" type="checkbox"/> | 0 500n conservative |
| 2 dc | <input type="checkbox"/> | t |

Outputs

| Name/Signal/Expr | Value | Plot | Save | Save Opti |
|------------------|----------|-------------------------------------|-------------------------------------|-----------|
| 1 out<1280> | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | yes |
| 2 tsub | 18.8061n | <input checked="" type="checkbox"/> | <input type="checkbox"/> | |
| 3 in | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | no |
| 4 net13 | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | no |

Plot after simulation: **None** Plotting mode: **Append**

Status: Ready | T=27 C | Simulator: spectre_aps | State: tsub_deff

Anexo 5. Tiempo de subida mínimo del buffer.

The screenshot displays the Cadence ADE L (14) interface for a simulation setup. The window title is "ADE L (14) - practice_3v0_hwg_test_dc_inv_cmos_tx schematic (on ndesktop14)". The menu bar includes "Launch", "Session", "Setup", "Analyses", "Model(AF)", "Variables", "Outputs", "Simulation", and "Results". The toolbar shows various icons for file operations and simulation control.

The **Design Variables** panel shows a table with the following data:

| Name | Value |
|-----------|-------|
| 1 avdd_dc | 5 |

The **Analyses** panel shows a table with the following data:

| Type | Enable | Arguments |
|------|-------------------------------------|-----------|
| 1 dc | <input checked="" type="checkbox"/> | t |

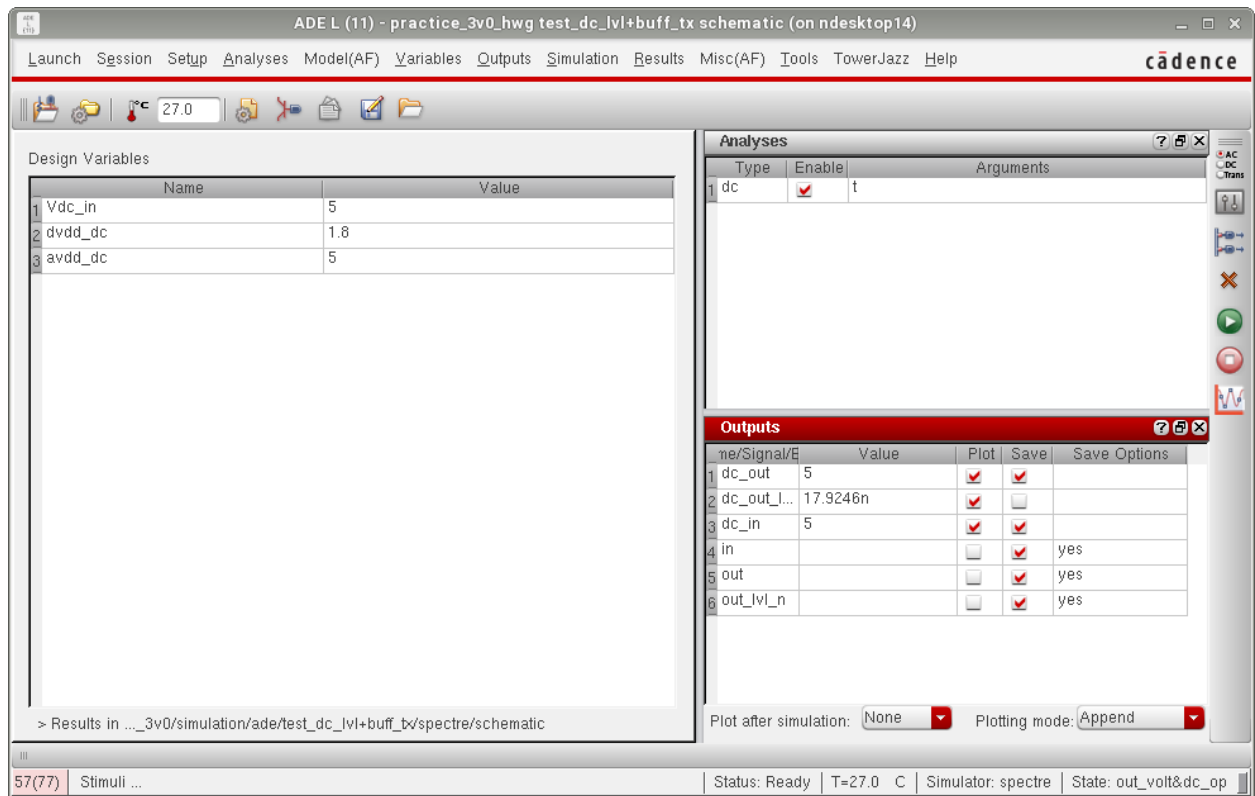
The **Outputs** panel shows a table with the following data:

| Name/Signal/Expr | Value | Plot | Save | Save Opt |
|------------------|----------|-------------------------------------|-------------------------------------|----------|
| 1 dc_in | 5 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
| 2 dc_out | 22.9254n | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
| 3 in | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | yes |
| 4 out | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | yes |

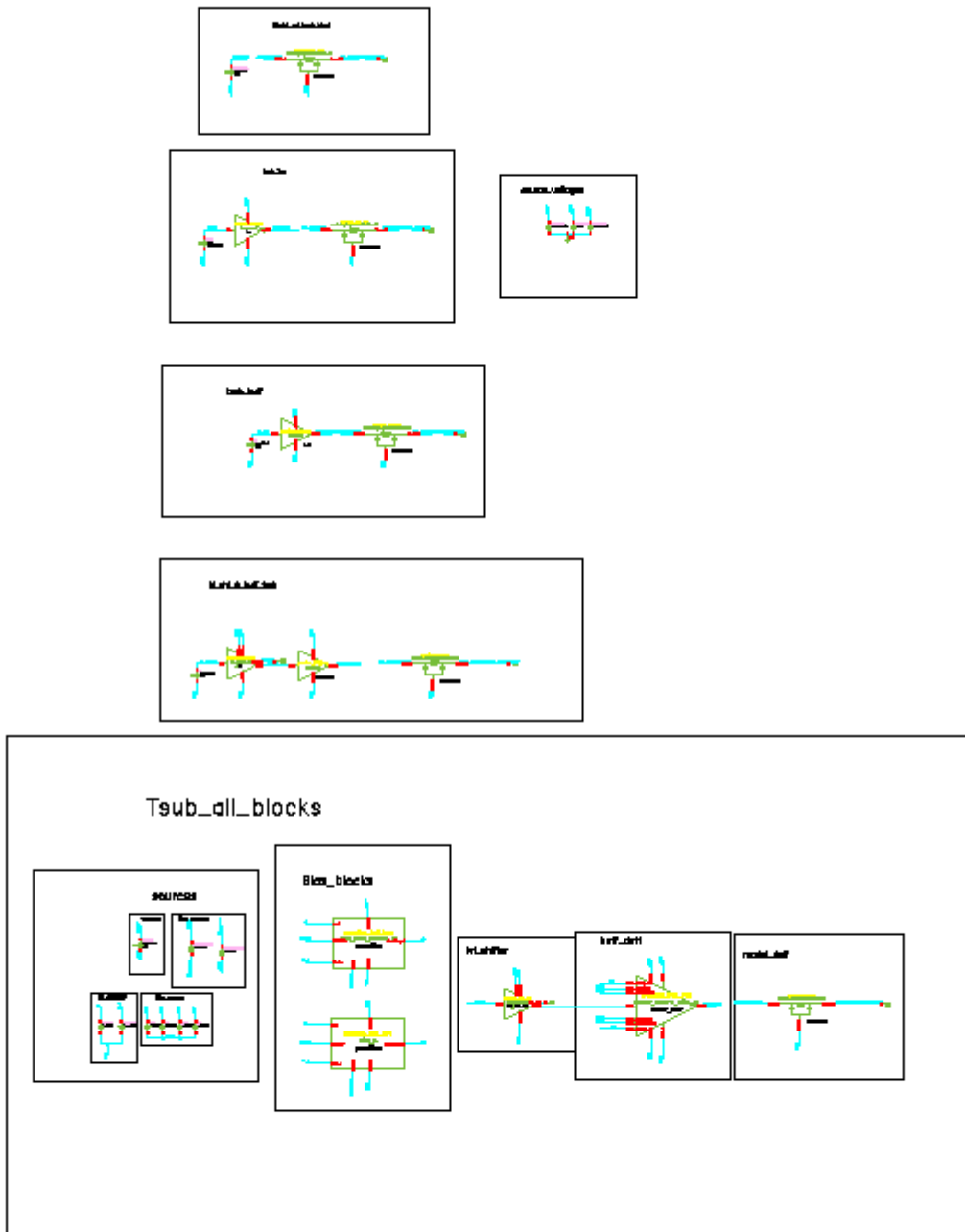
At the bottom of the Analyses panel, there are dropdown menus for "Plot after simulation:" (set to "None") and "Plotting mode:" (set to "Append").

The status bar at the bottom shows: "63(86) | Load State ... | Status: Ready | T=27 C | Simulator: spectre | State: spectre_state1".

Anexo 6. Test DC inverter CMOS.



Anexo 7. Test DC level shifter más buffer.



Anexo 8. Esquemático de todas las pruebas de tiempo de subida.

BIBLIOGRAFÍA

- [1] D. X. D. Yang, *Digital Pixel CMOS Image Sensors*, vol. 2, 1999.
- [2] M. C.-K. C. a. V. P. C. Albert H. Titus, «CMOS photodetectors.», *University of Montreal*, 2011.
- [3] K. Vetenskapsakademien, «The Nobel Priza in physics 2009,» 2009.
- [4] Scientific Imagin Technologies, Inc, «SITe CCD Technology for Superior Perfomance.,» *An Introduction to Scientific Imagin Charge-Coupled Devices.*, 1994.
- [5] D. A. a. J. W. S. R.S. Aikens, «Solid-State Imagers for Microscopy,» 1989, p. Chapter 16.
- [6] U. o. C. Ann Holms, «Complementary Metal-Oxide Semiconductor Sensors,» Santa Barbara, 2010.
- [7] ANDOR an Oxford Instruments company, «Electron Multiplying CCD Cameras».
- [8] J. F. J. S. M. igasm E. Cabruja, *Review of CMOS image sensors*, Campus Universitat Autònoma de Barcelona.: Centre Nacional de microelectrónica, 2006.
- [9] DALSA Digital Cinema., *Image Sensor Architectures for Digital Cinematogrpahy*, Waterloo, Ontario, Canada.
- [10] N. Blanc, *CCD versus CMOS- has CCD imaging come to an end?*, Zurich, 2001.
- [11] A. Dickinson, B. Ackland, E. Eid y D. Inglis, *Standar CMOS Active Pixel Image Sensors for Multimedia Applications*, 1995.
- [12] R. Coath, J. Crooks, A. Godbeer, M. Wilson, R. Turchetta y S. Collaboration, *Advanced Pixel Architectures for Scientific Image Sensors*, Turherford Appleton Laboratory, UK..
- [13] D. J. Linkemann y B. Weber, «White Paper, Global Shutter, Rolling Shutter- Functionality and Characteristics of Two Exposure Methods».
- [14] M. Hansard, S. Lee, O. Choi y R. Horaud, «Time of flight Cameras: Principles, Methods, and Applications,» 2012.
- [15] Texas Instrument, «Introduction to the Time-of-Flight (ToF) System Desing. User's Guide,» 2013.
- [16] ST.Life. augmented, «Optical Sensing: 1D to 3D using Time-of-Flight Technology.,» 2013.
- [17] D. Stoppa, «3D Imaging with ToF: Solid-State Devices, Circuits and Architectures.,» 2015.
- [18] S.-E. Kim y D. L. Parker, «Time-of-Flight Angiography».
- [19] N. N. Victor Castaneda, «Time-of-Flight and Kinect Imaging,» *Kinect Programming for computer Vision*,

2011.

[20] B. Razavi, Desing of Analog CMOS Integrated Circuits, McGraw-Hill International Edition, 2001.