

Diseño de Conversores Analógicos Digitales

Clase 3 - Comparator

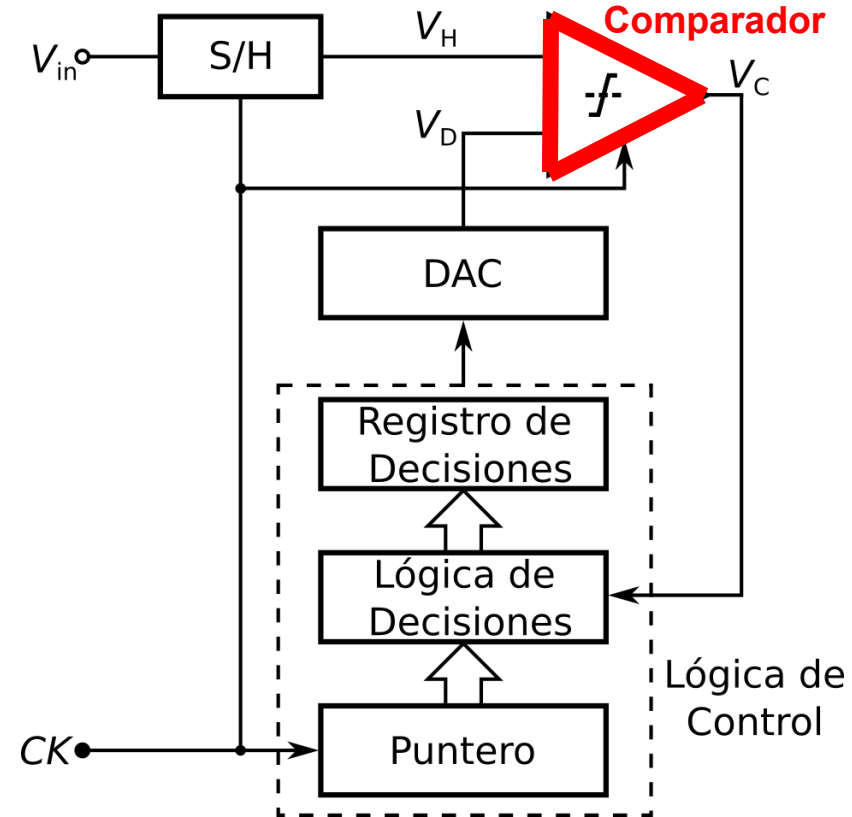
FIUBA, Argentina

- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

- **Introducción**
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

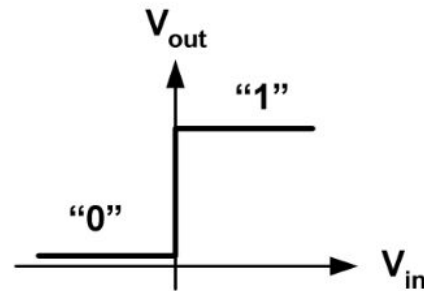
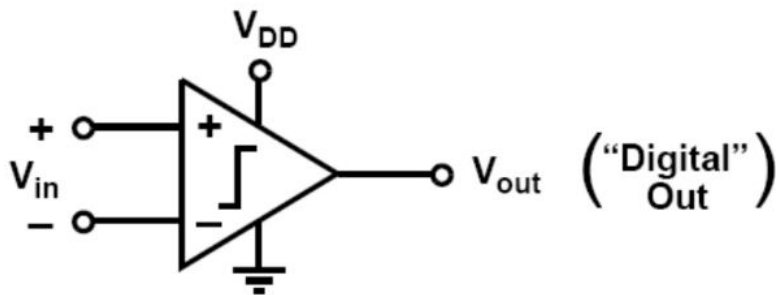
Introducción

- En ADCs de tipo SAR, flash, pipeline, etc, se utilizan comparadores para decidir si un voltaje analógico de entrada corresponde a un 0 o un 1 lógico
- Particularmente en ADCs de tipo SAR, el comparador es uno de los circuitos más importantes ya que define el consumo, la velocidad y el ruido de la conversión



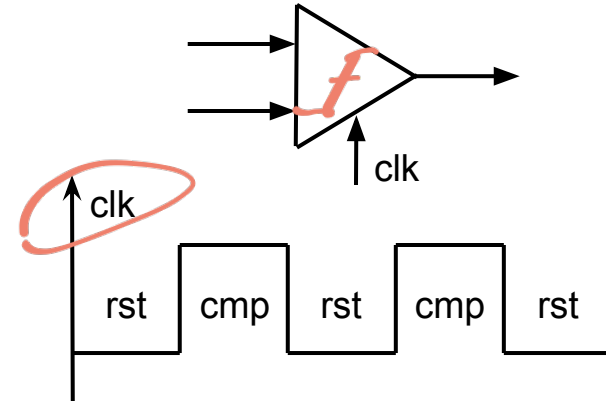
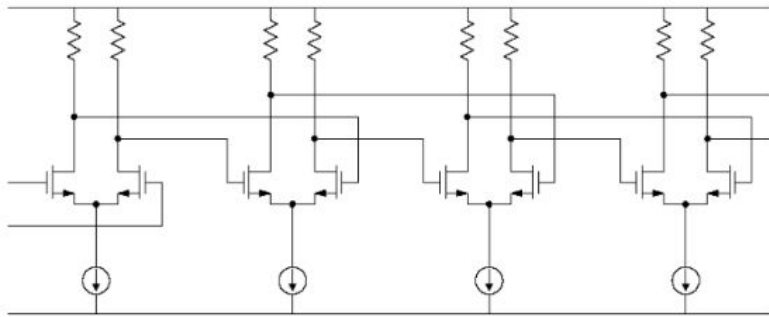
Introducción - Comparador de tensión ideal

- Como se mencionó anteriormente, su función es comparar dos voltajes analógicos de entrada y generar un resultado digital (0 o 1) dependiendo de cuál entrada es de mayor magnitud
- Idealmente un comparador de voltaje tiene ganancia infinita y genera el resultado de la comparación de manera instantánea



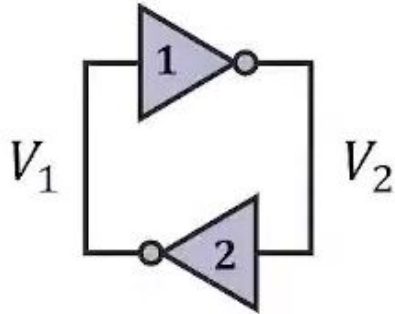
Introducción - Comparador estático y dinámico

- En la bibliografía existen una gran cantidad de comparadores, que se pueden clasificar según su operación sea estática o dinámica
- Los comparadores estáticos están basados tradicionalmente en amplificación por par diferencial
 - Consumen potencia estática ya que están constantemente en funcionamiento
- Los comparadores dinámicos dependen de una señal de reloj y tienen dos etapas de funcionamiento: comparación y reset
 - Basados en latch regenerativos que no consumen potencia estática

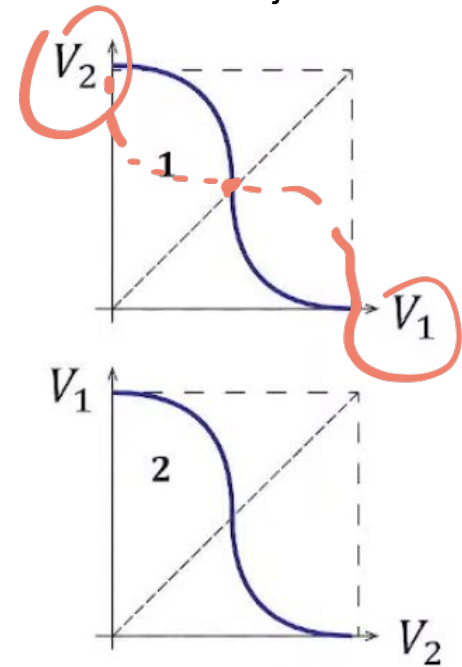


- Introducción
- **Latch CMOS**
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

- El latch CMOS, también llamado latch regenerativo, consiste en dos inversores conectados en realimentación positiva
- El circuito no posee entrada
- Tiene 2 estados estables y un estado metaestable



Transferencia estática de voltaje.



- El latch regenerativo convergerá a uno de los estados estables según sus condiciones iniciales

Circuito esquemático CMOS

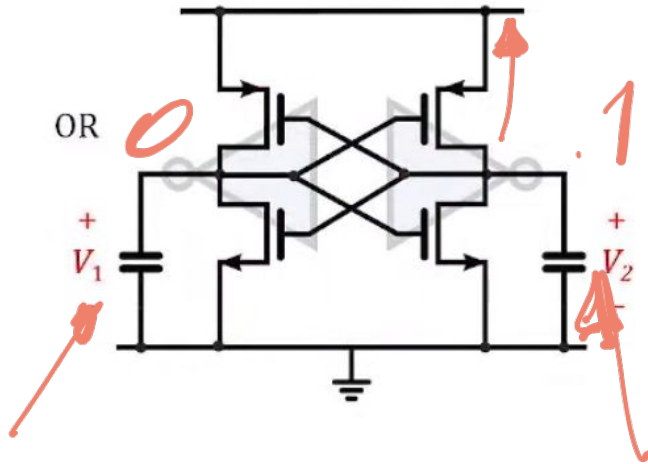
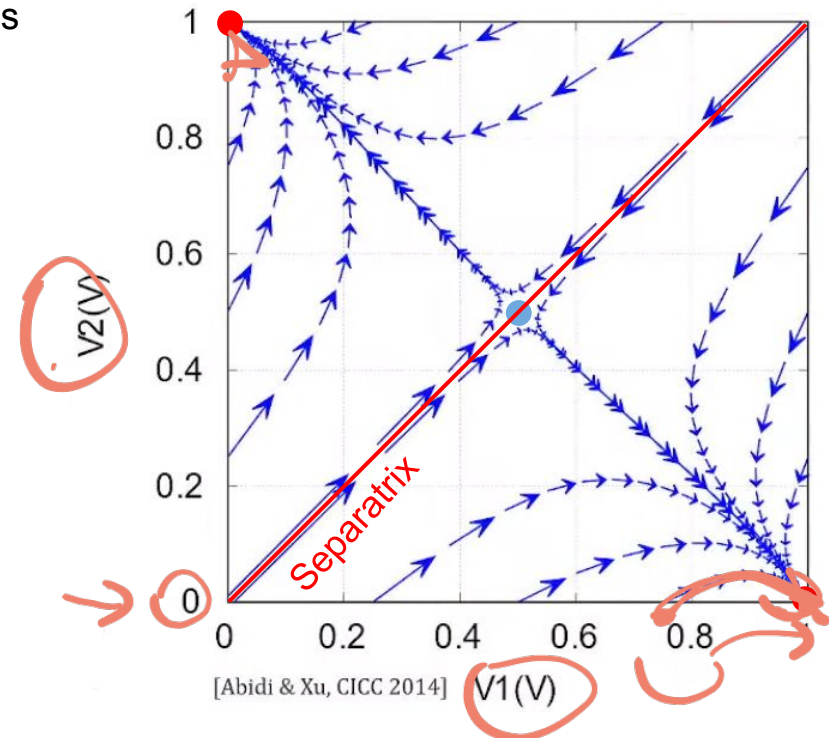
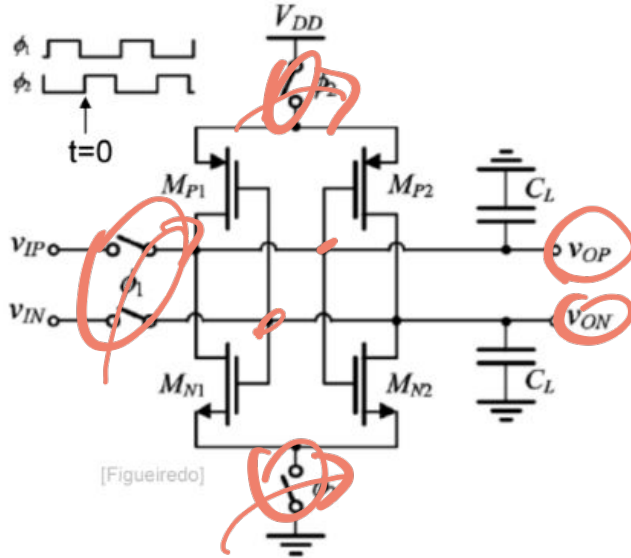


Diagrama de estado temporal



Latch CMOS - Ganancia

Conceptual Circuit



- phi1: Set up initial condition (v_{OD0})
- phi2: Enable positive feedback

Inverter Transconductance

$$G_m = g_{mN} + g_{mP}$$

$$\frac{dv_{OP}}{dt} = \frac{i_1(t)}{C_L} = -\frac{G_m v_{ON}(t)}{C_L}$$

$$\frac{dv_{ON}}{dt} = \frac{i_2(t)}{C_L} = -\frac{G_m v_{OP}(t)}{C_L}$$

$$\tau = \frac{G_m}{C_L}$$

$$v_{OD}(t) = v_{OP}(t) - v_{ON}(t)$$

$$v_{OD}(t) = v_{OD0} \cdot e^{t/\tau}$$

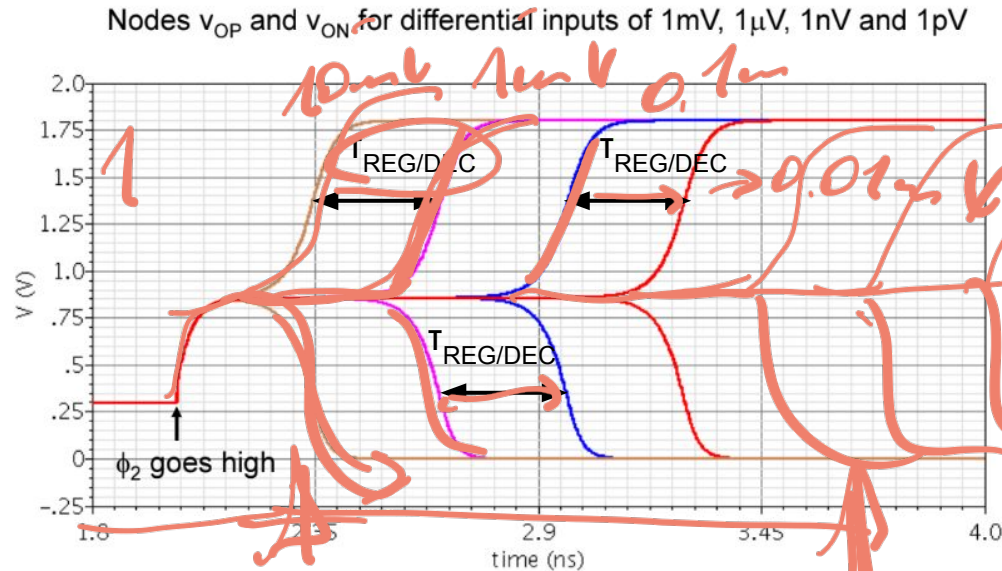
$$A(t) = \frac{v_{OD}(t)}{v_{OD0}} = e^{t/\tau}$$

Ganancia del latch vs tiempo

$A(\tau_d)$	τ_d/τ
10	2.3
100	4.6
1,000	6.9
10,000	9.2

- Donde τ_{REG} corresponde al tao de regeneración.

- Una magnitud interesante es el $\tau_{\text{REG/DEC}}$, cuyo valor indica cuánto es la diferencia de tiempo de regeneración entre dos entradas que difieren en una década.



- La relacion entre el τ_{REG} y $\tau_{\text{REG/DEC}}$ está dado por:

- $V_{\text{OD1}} = A_v * V_{\text{in}} e^{t1/\tau_{\text{REG}}}$
- $V_{\text{OD2}} = A_v * (V_{\text{in}}/10) e^{t2/\tau_{\text{REG}}}$
- $V_{\text{OD1}} = V_{\text{OD2}}$
- $A_v * V_{\text{in}} e^{t1/\tau_{\text{REG}}} = A_v * (V_{\text{in}}/10) e^{t2/\tau_{\text{REG}}}$
- $e^{t1/\tau_{\text{REG}}} = (1/10) e^{t2/\tau_{\text{REG}}}$
- $t1/\tau_{\text{REG}} = \ln(1/10) + t2/\tau_{\text{REG}}$
- $t1/\tau_{\text{REG}} = -\ln(10) + t2/\tau_{\text{REG}}$
- $\ln(10) = (t2-t1)/\tau_{\text{REG}}$

$$\tau_{\text{REG/DEC}} = \ln(10) * \tau_{\text{REG}}$$

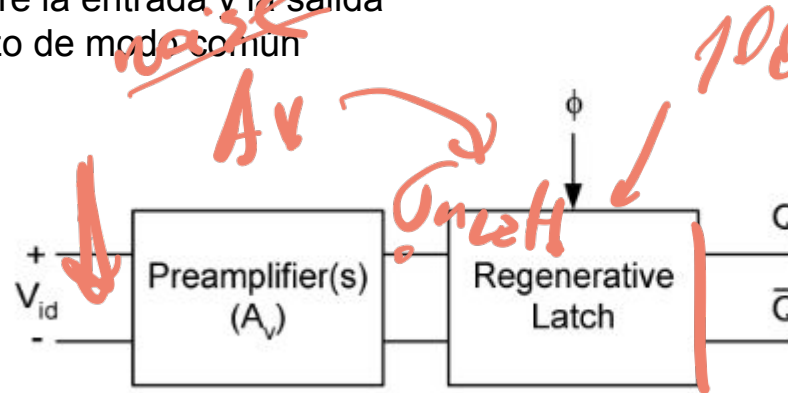
$$\tau_{\text{REG}} = \tau_{\text{REG/DEC}} / \ln(10)$$

- Tener en cuenta:

- $\ln(a+b) = \ln(a) + \ln(b)$
- $\ln(0.1) = -\ln(10)$
- $\tau_{\text{REG/DEC}} = t2-t1$

Latch CMOS - Preamplificación

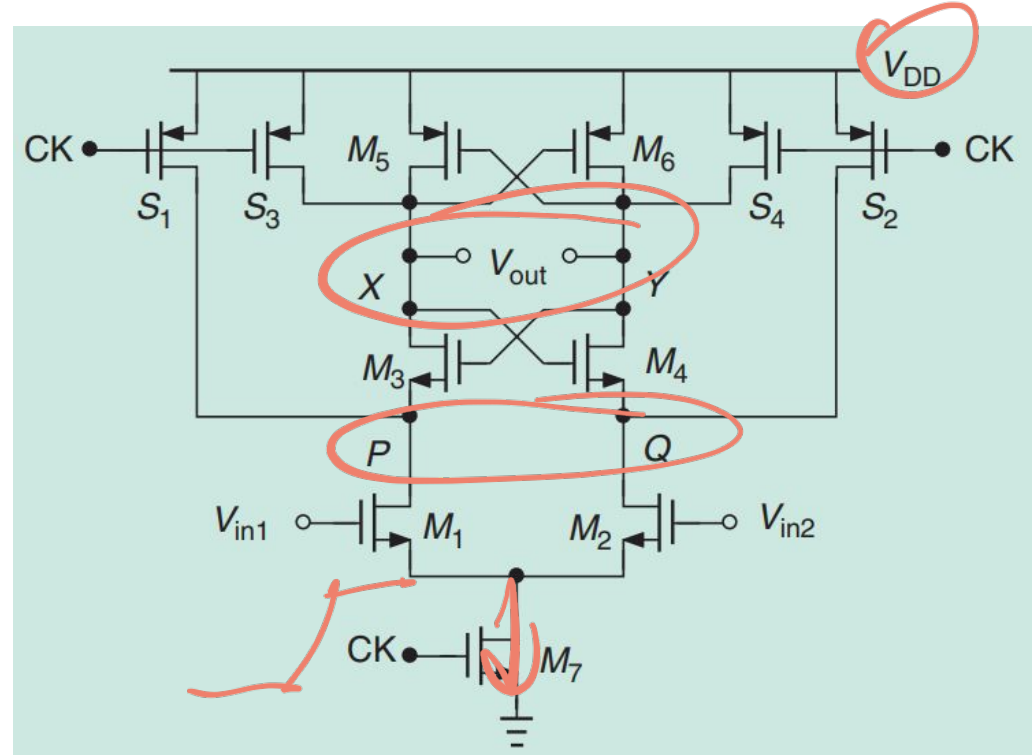
- Los latch regenerativos típicamente son precedidos por una etapa de preamplificación, es decir que se amplifica ligeramente la señal de entrada antes de ser regenerada
- Se obtienen las siguientes ventajas:
 - Menor offset de entrada (el offset del latch regenerativo tiende a ser grande)
 - Menor ruido térmico
 - Menor tiempo de regeneración
 - Atenuación del kickback noise
 - Aislación entre la entrada y la salida
 - Mejor rechazo de modo común



- Introducción
- Latch CMOS
- **Comparadores tradicionales:**
 - **Strong Arm**
 - **Double Tail**
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

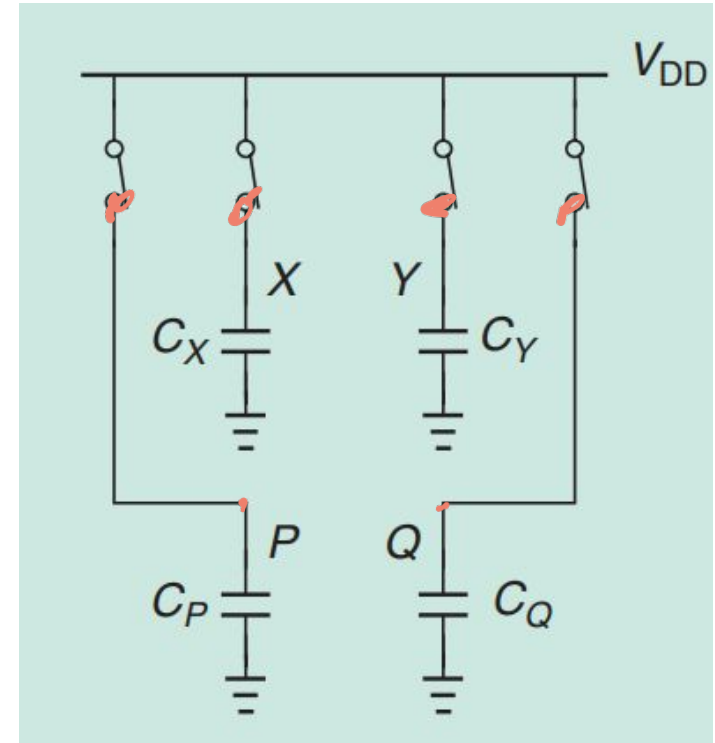
Strong Arm Comparator

- Es un comparador dinámico muy utilizado en la actualidad
- Ventajas:
 - No consume potencia estática
 - La excursión de salida es rail to rail
 - Su offset y ruido referido a la entrada dependen principalmente del par diferencial
 - Baja complejidad de diseño y operación
- Sufre kickback por el acoplamiento entre la entrada y otros nodos de gran excursión.



Strong Arm Comparator - Operación (1/4)

- Su funcionamiento se divide en 4 etapas
- Etapa 1:
 - Cuando la señal de reloj está en alto ($CK = 1$)
 - El comparador está reseteado y los nodos P,Q y X,Y son llevados a VDD

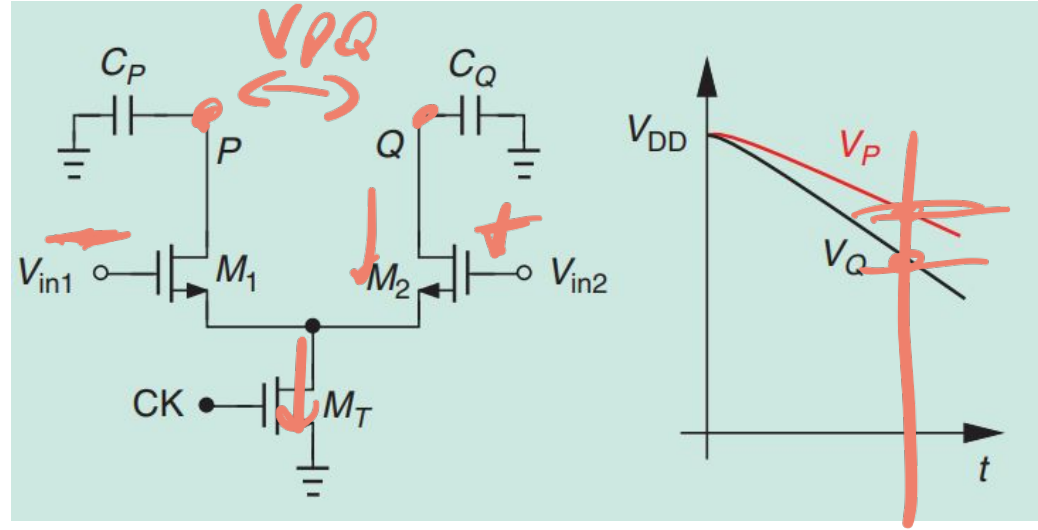


Strong Arm Comparator - Operación (2/4)

- Etapa 2:

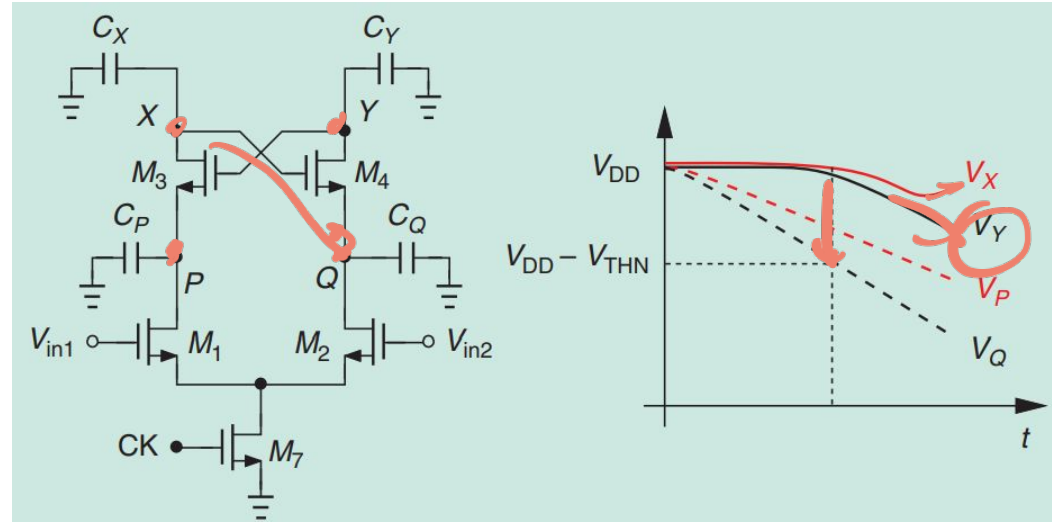
- La señal de reloj está en bajo (CK = 0)
- Las llaves S1-4 se abren, M7 se cierra y comienza a circular una corriente diferencial por el par diferencial M1-2
- Los nodos P,Q comienzan a descargarse de manera diferencial proporcional a $V_{in1} - V_{in2}$
- La tensión P-Q puede superar a $V_{in2} - V_{in1}$, por lo que existe una amplificación diferencial

$$A_v \approx \frac{g_{m1,2} V_{THN}}{I_{CM}}$$



Strong Arm Comparator - Operación (3/4)

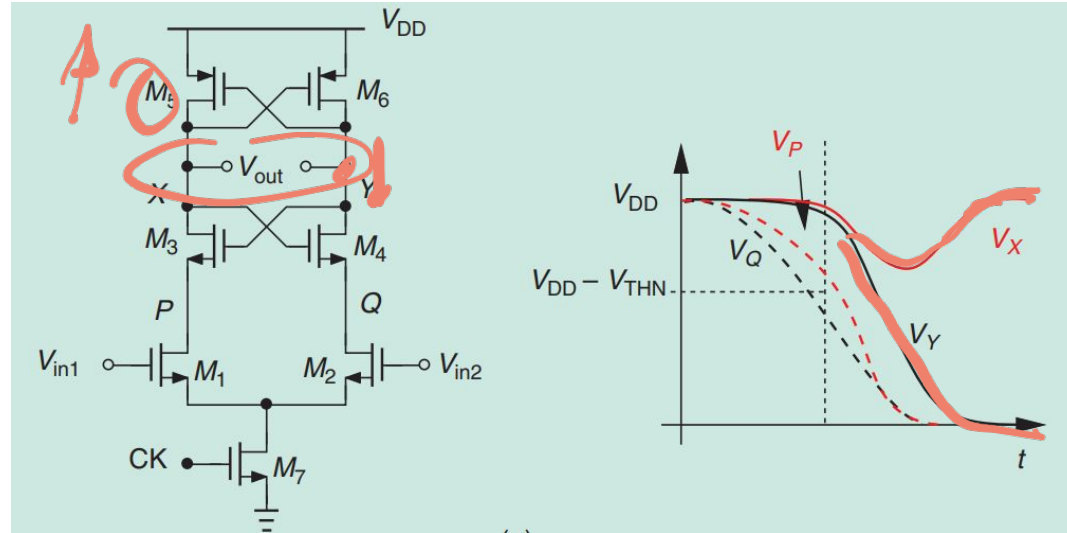
- Etapa 3:
 - Cuando los nodos P,Q se descargan por debajo del umbral de M3-4, estos se encienden comenzando a descargar los nodos X,Y



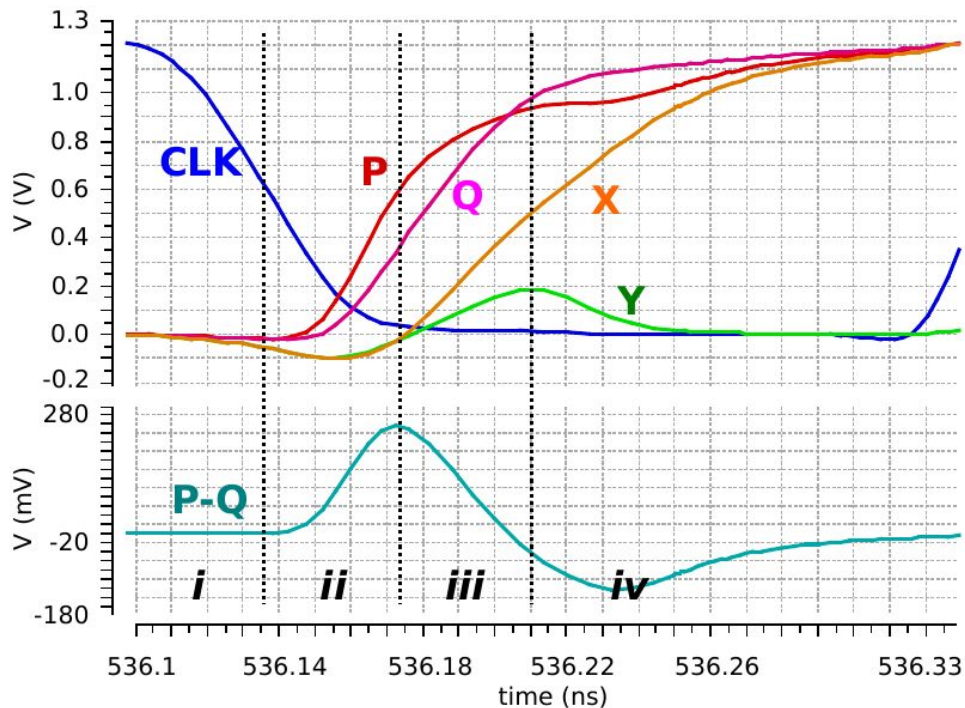
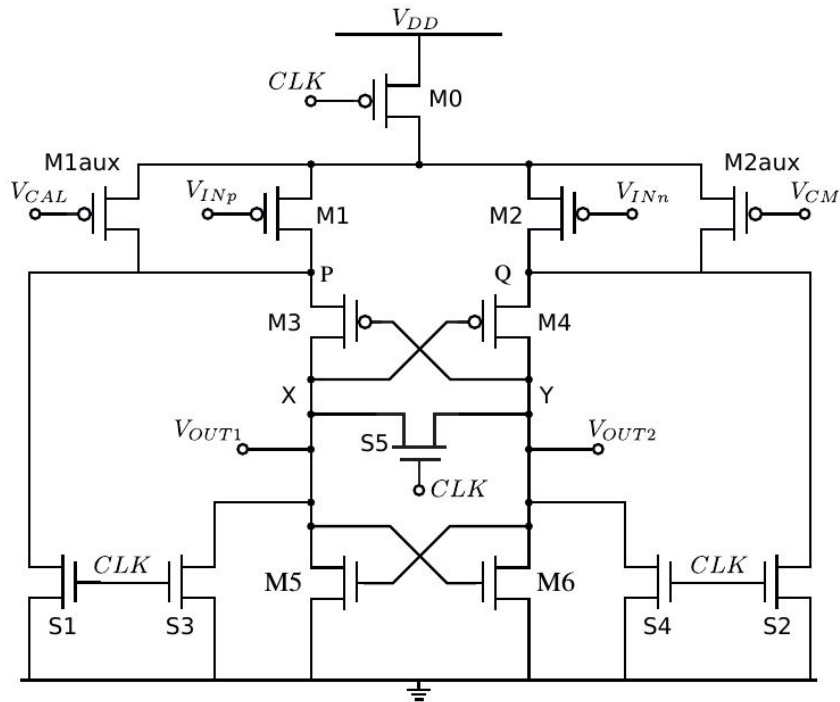
Strong Arm Comparator - Operación (4/4)

- Etapa 4:

- Cuando los nodos X,Y se descargan debajo del umbral de M5-6, estos se encienden y se crea una realimentación positiva entre los transistores M3-6
- Esta realimentación lleva uno de los nodos X o Y a VDD y el otro a tierra
- Este estado se mantiene hasta que una nueva señal de reloj (CK = 1) habilite la siguiente comparación

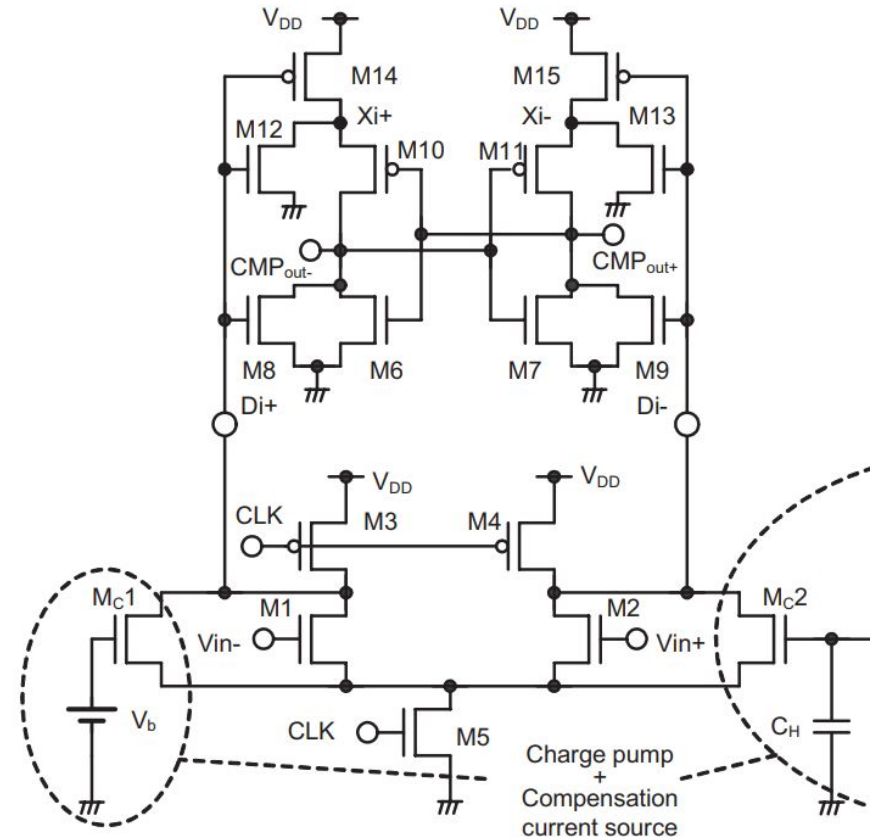


Strong Arm Comparator - Operación completa



Double Tail Comparator

- Se compone de dos etapas en cascada
 - Amplificador: Provee ganancia a la señal de entrada
 - Latch regenerativo: regenera (a 0 o 1) en base a la señal amplificada
- La entrada (baja excursión) no está acoplada directamente a la salida (gran excursión), por lo que no sufre kickback



- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- **Análisis de ruido**
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

- La mayor parte del ruido referido a la entrada se origina en M1 y M2
- Los otros transistores entran en juego sólo después de que se ha acumulado una ganancia significativa, reduciendo su aporte de ruido
- En la etapa de amplificación, el circuito se convierte en un integrador generando ruido de salida debido a M1 y M2

$$\sigma_{1,2}^2 = \frac{8kT\gamma}{C_{P,Q}} \cdot \frac{g_{m1,2} V_{THN}}{I_{CM}} \cdot$$

- Teniendo en cuenta que

$$\left(A_v \approx \frac{g_{m1,2} V_{THN}}{I_{CM}} \right)$$

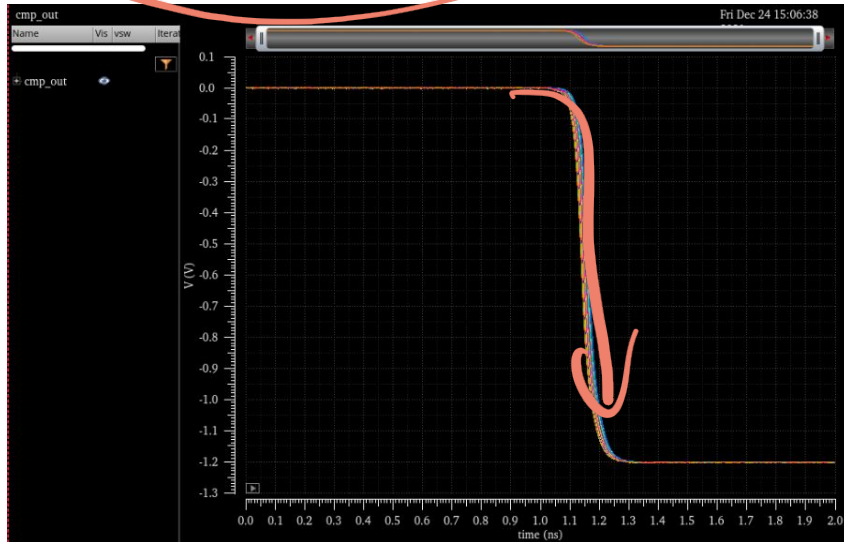
- Luego el ruido referido a la entrada es

$$\left[\overline{V_{n,in}^2} = \frac{8kT\gamma}{A_v C_{P,Q}} \right]$$

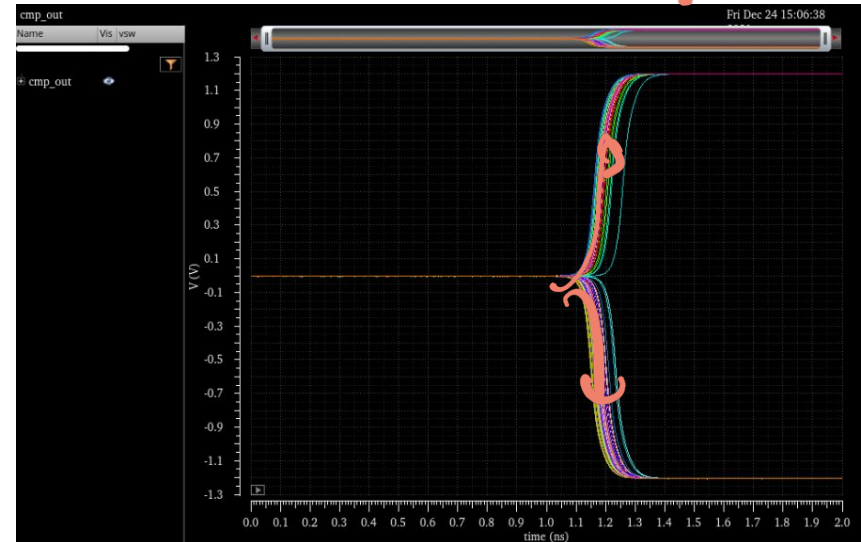
Análisis de Ruido - Configuración de test bench

- Se realiza un análisis transient noise de un solo ciclo de duración, y muchas repeticiones (~100 simulaciones)
- En cada corrida el comparador decide por 1 o por 0 según sea el valor de entrada
- Se itera el proceso anterior para diferentes valores de entrada alrededor del umbral

• $V_{in} = -1.8\text{mV}$

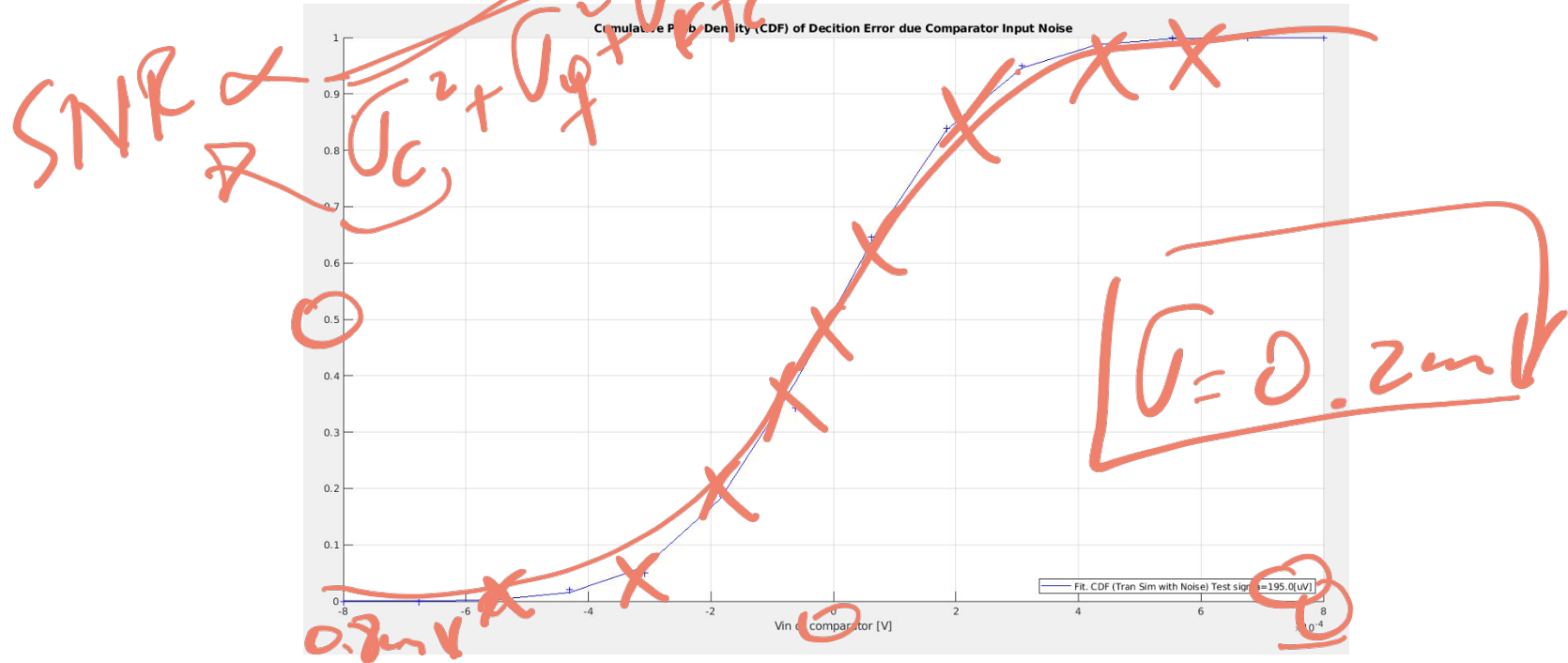


• $V_{in} = 0.13\text{mV}$



Análisis de Ruido - Configuración de test bench

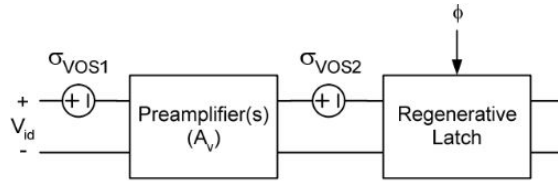
- Promediando los valores de salida obtenidos (V_{out}) para cada V_{in} , se obtiene la curva CDF vs V_{in} .
- Finalmente por script se obtiene el valor final de sigma de ruido.



- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- **Análisis de offset**
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- Trabajo práctico

Análisis de Offset

- El error de offset puede entenderse como un desplazamiento del umbral de decisión, y es originado por los desapareamientos entre las ramas diferenciales
- Existen 2 tipos de offsets:
 - ~~Sistemático~~: es debido a un diseño defectuoso de layout y puede corregirse por el diseñador
 - Aleatorio: es introducido por el proceso de fabricación e imposible de corregir por diseño
- De manera similar al análisis de ruido, la mayor parte del offset referido a la entrada se origina en M1 y M2 ya que los desapareamientos entre los otros transistores son divididos por la ganancia de amplificación

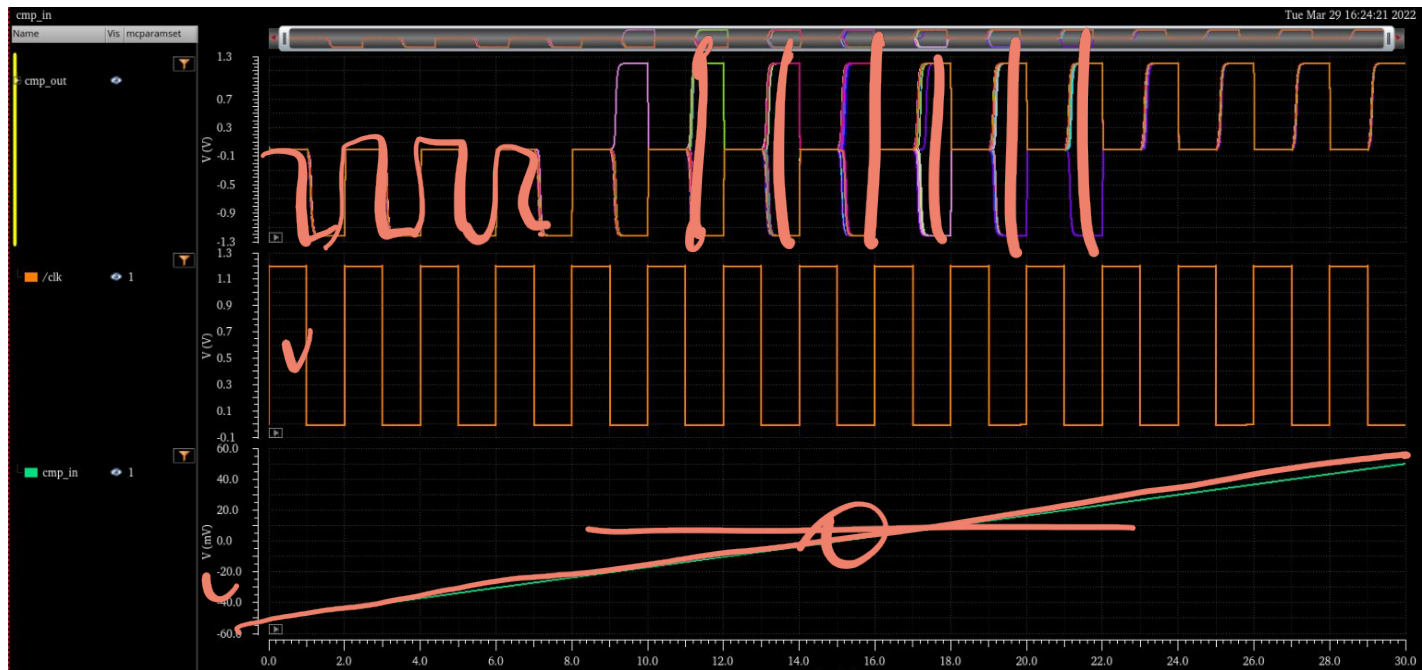


$$\sigma_{VOS}^2 = \sigma_{VOS1}^2 + \frac{1}{A_v^2} \sigma_{VOS2}^2$$

- El error de offset podría corregirse introduciendo una asimetría en el comparador, modificando de manera intencional el umbral de decisión para contrarrestar el offset original

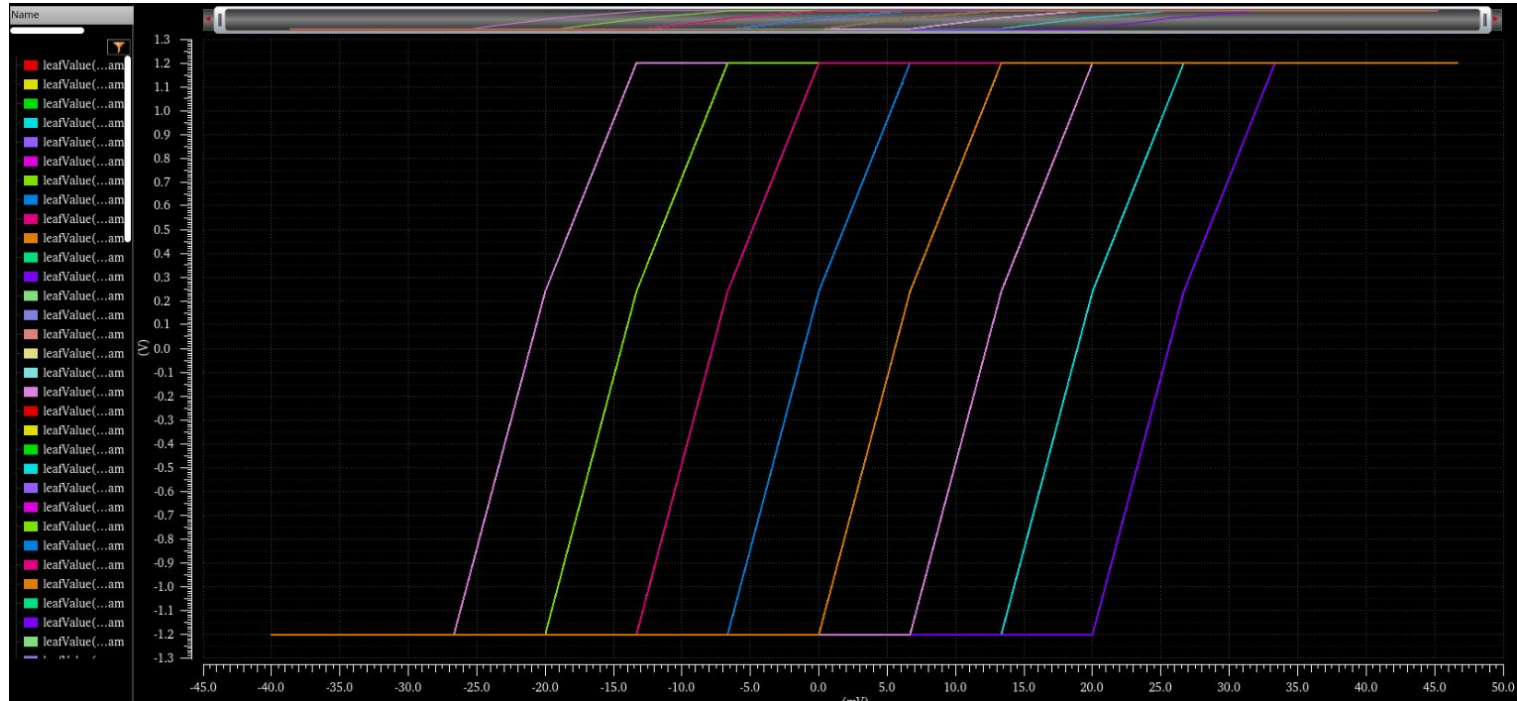
Análisis de Offset - Configuración de test bench

- Se realiza un test montecarlo con mismatch (Setear corner montecarlo)
- Cada corrida de montecarlo consiste en un transient donde se barre la entrada del comparador (con una rampa de entrada o un barrido paramétrico)



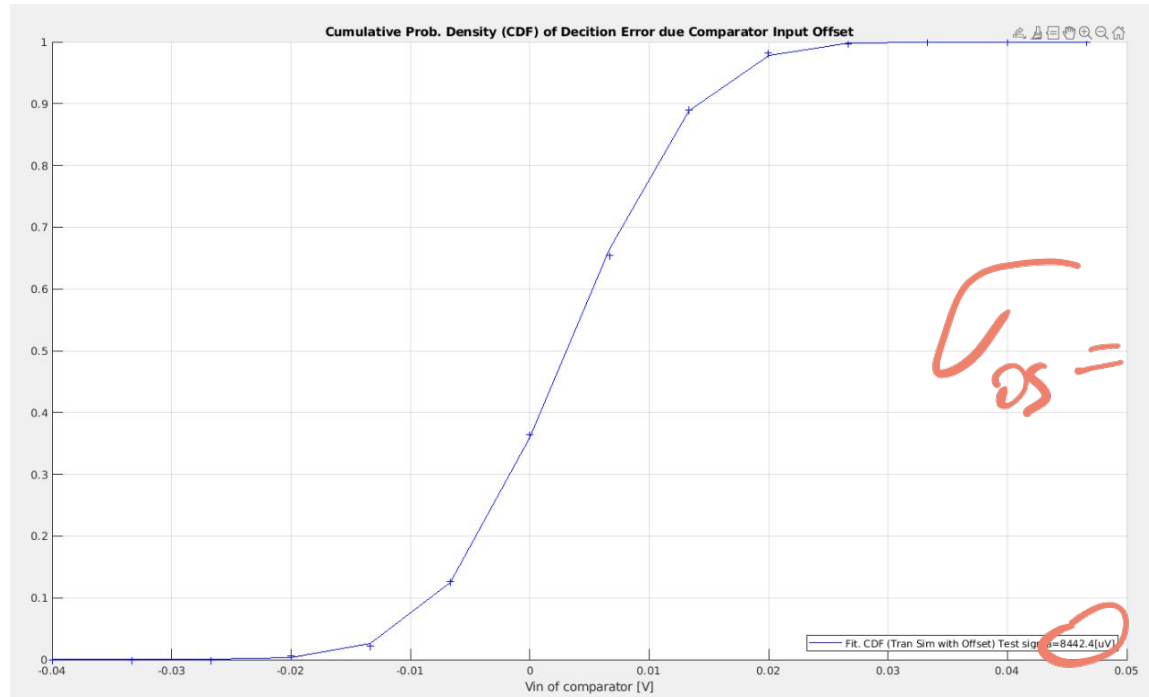
Análisis de Offset - Configuración de test bench

- Se obtiene una curva V_{out} vs V_{in} para todas las corridas



Análisis de Offset - Configuración de test bench

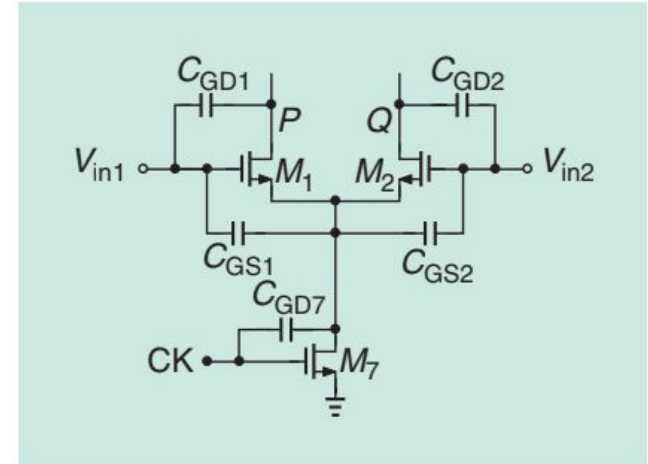
- Haciendo un promedio de la curva V_{out} vs V_{in} se obtiene la curva CDF vs V_{in}
- Finalmente por script se obtiene el valor final de sigma de offset



- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- **Problemas:**
 - **Kickback Noise**
 - **Metaestabilidad**
 - **Histéresis**
- Lazo de autoclock
- Trabajo práctico

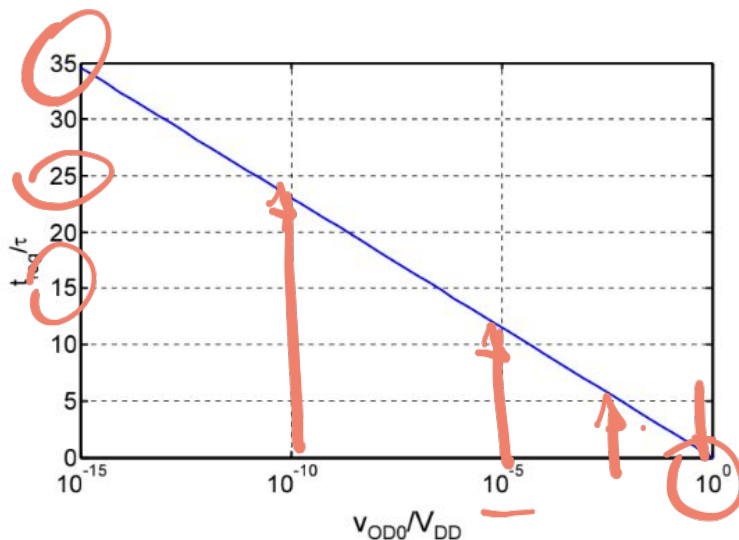
Kickback Noise

- El kickback noise se produce las por altas corrientes de transición que existen entre las ramas de salida y las entradas del comparador, exhibiendo componentes diferenciales y de modo común
- La componente diferencial aparece cuando V_p y V_q se descargan a 0 (a distinta velocidad) y acoplan con la entrada a través de C_{GD1} y C_{GD2} , descargando las ramas de entrada de manera diferencial
- La componente de modo común es debido al acople de la señal de reloj (CK) con las entradas, por medio de $C_{GS1,2}$ y C_{GD7} .



Metaestabilidad

- La metaestabilidad es un problema que presenta un latch regenerativo cuando las entradas están próximas entre sí
- A medida que es menor la entrada diferencial inicial del latch regenerativo, la salida tardará más tiempo en ser regenerada por el latch



Tiempo necesario para regenerar una determinada entrada diferencial v_{OD0}

$$\frac{t_{reg}}{\tau} = \ln\left(\frac{V_{DD}}{v_{OD0}}\right)$$

- ¿Qué sucede si la entrada diferencial es tan pequeña que la regeneración no se produce a tiempo?
 - Esto es llamado estado metaestable. Las salidas del comparador permanecen unidas en un estado intermedio entre “1” y “0” lógicos, lo que puede producir un mal funcionamiento del conversor si no se trata dicho estado adecuadamente (en la unidad de lógica se profundiza sobre esto)
- ¿Cómo evitar un estado metaestable?
 - No es posible evitar un estado metaestable, simplemente debe darse suficiente tiempo al comparador para que pueda finalizar la decisión correctamente (en la unidad de lógica se profundiza sobre esto)
- ¿Por qué NO usar detectores de metaestabilidad?
 - Una opción para “salir” de un estado metaestable sería intentar detectarlo. Sin embargo, es muy probable que en el proceso de detectar la metaestabilidad el comparador se decida, inutilizando el algoritmo de detección.

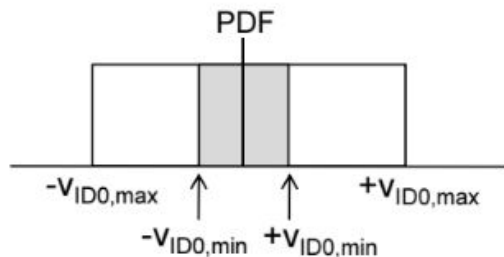
- Asumir que el tiempo máximo de regeneración es t_{\max} , entonces la entrada diferencial mínima del latch debe ser

$$A(t) = \frac{v_{OD}(t)}{v_{OD0}} = e^{t/\tau} \longrightarrow v_{OD0,min} = \frac{V_{DD}}{e^{t_{\max}/\tau}}$$

- Si tuviéramos un circuito de preamplificación

$$v_{ID0,min} = \frac{1}{A_v} \frac{V_{DD}}{e^{t_{\max}/\tau}}$$

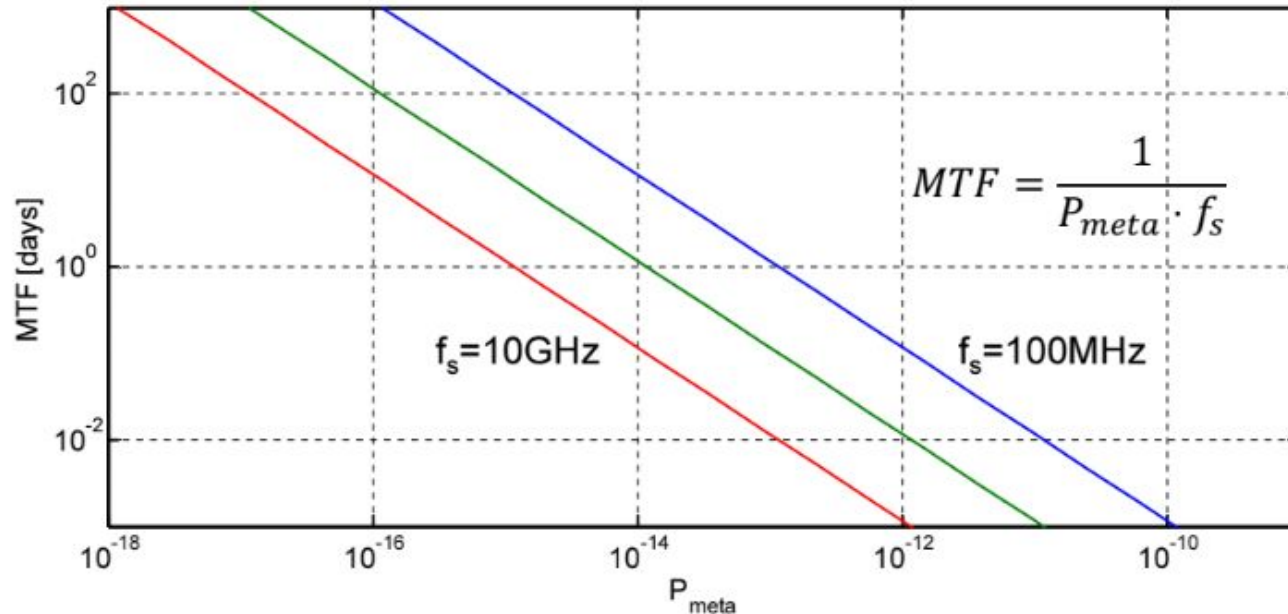
- Ahora asumir un entrada uniformemente distribuida sobre todo el rango de entrada, la probabilidad de que ocurra un estado metaestable es



$$P_{meta} = \frac{v_{ID0,min}}{v_{ID0,max}} = \frac{V_{DD}}{v_{ID0,max}} \frac{e^{-t_{\max}/\tau}}{A_v}$$

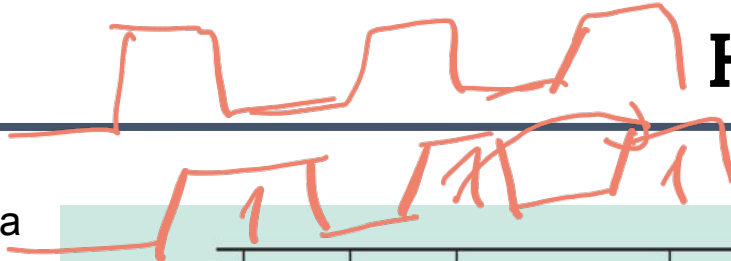
Metaestabilidad

- Es útil entender la metaestabilidad como la media de tiempo entre estados metaestables (MTF, *mean time to failure*)



A hand-drawn diagram illustrating a step function. The function is plotted on a coordinate system with a horizontal axis and a vertical axis. The function is zero for $x < 0$ and for $x > 1$. Between $x = 0$ and $x = 1$, the function has three steps: it jumps to 1 at $x = 0$, drops to 0 at $x = 0.5$, and jumps back to 1 at $x = 0.75$. The area under the curve is shaded in light blue. The label 'a' is at the bottom left, and 'b' is at the top right.

-
- A hand-drawn diagram illustrating a step function. The function is plotted on a coordinate system with a horizontal axis and a vertical axis. The function is zero for $x < 0$ and for $x > 1$. Between $x = 0$ and $x = 1$, the function has three steps: it jumps to 1 at $x = 0$, drops to 0 at $x = 0.5$, and jumps back to 1 at $x = 0.75$. The area under the curve is shaded in light blue. The label 'a' is at the bottom left, and 'b' is at the top right.



- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- **Lazo de autoclock**
- Trabajo práctico

- Introducción
- Latch CMOS
- Comparadores tradicionales:
 - Strong Arm
 - Double Tail
- Análisis de ruido
- Análisis de offset
- Problemas:
 - Kickback Noise
 - Metaestabilidad
 - Histéresis
- Lazo de autoclock
- **Trabajo práctico**

Trabajo Práctico (1/2)

- Diseñar un comparador Strong Arm y cumplir las siguientes especificaciones:

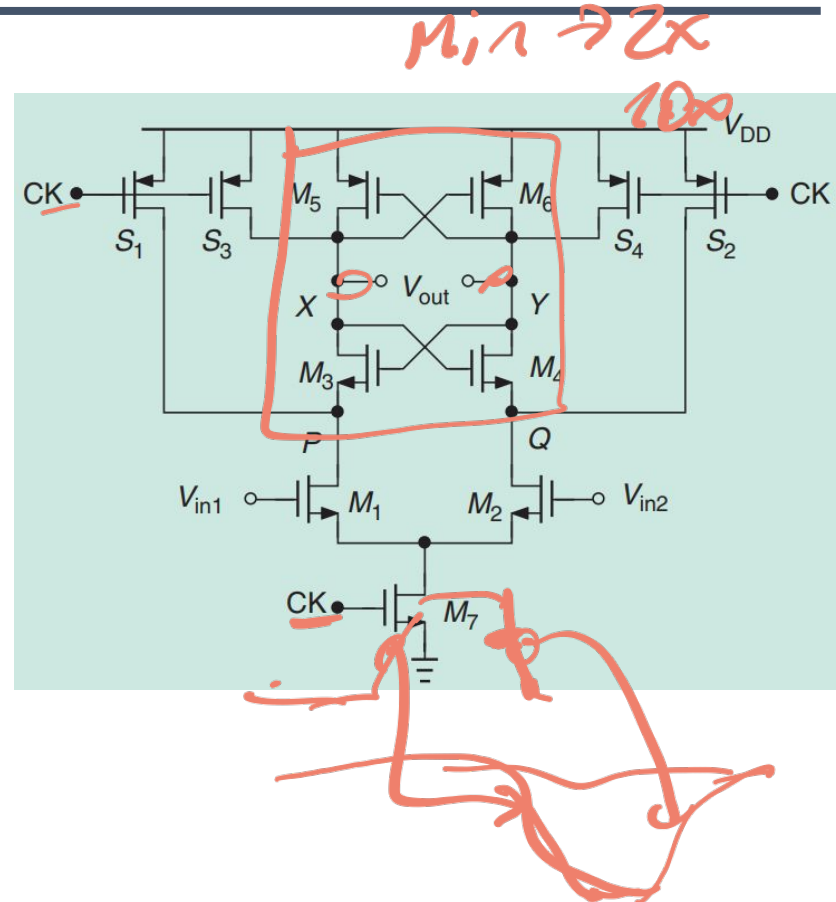
- Ruido referido a la entrada, $\sigma_{\text{cmp}} < 800\mu\text{V}$
- Velocidad de clock, $f_{\text{clk}} = 10\text{MHz}$ ($V_{\text{IN,diff}} = 10\text{mV}$).
- $C_L = 10\text{fF}$

- Medir los siguientes tiempos:

- Latencia de comparación (clk2reg)
- ~~Latencia de lazo reset (reg2clk)~~
- Latencia de reset de comparador (clk2reg)
- ~~Latencia de lazo habilitación (reg2clk)~~

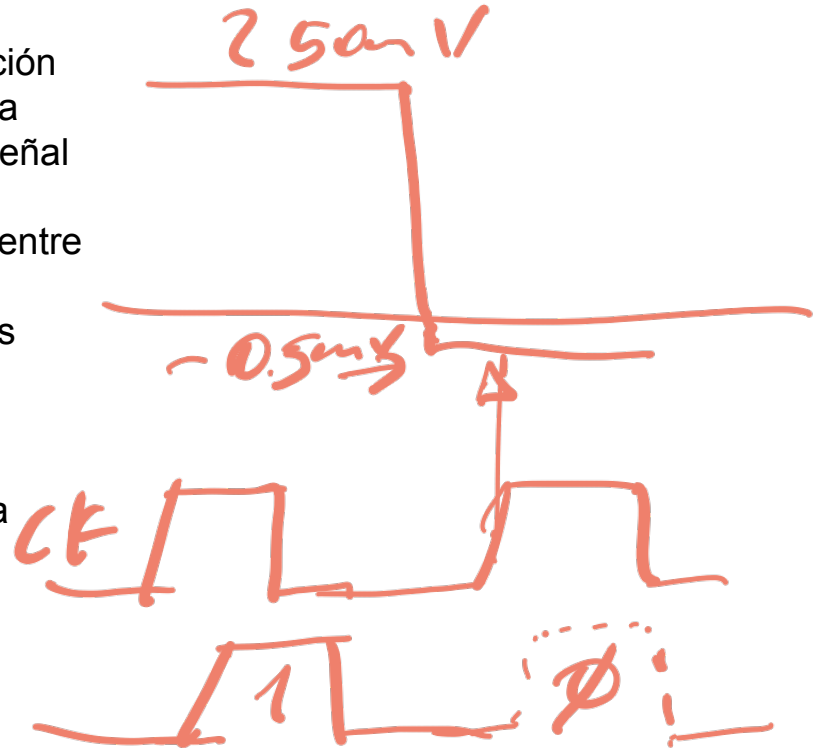
- Medir consumo en energía por ciclo (Joules/cycle)

- Medir offset de entrada al comparador diseñado



Trabajo Práctico (2/2) (OPCIONAL)

- Simular la histéresis del comparador diseñado.
- El test consiste en realizar una primera comparación con una señal de entrada en 250mVdiff, y luego la siguiente comparación debe realizarse con una señal de entrada de -0.5mVdiff.
 - El resultado del comparador debe cambiar entre ambas comparaciones
 - De esta manera se comprueba si el reset es capaz de eliminar toda la carga remanente debida a la primera comparación
- Repetir la simulación invirtiendo la polaridad de la señal de entrada (-250mVdiff a 0.5mVdiff)



FIN