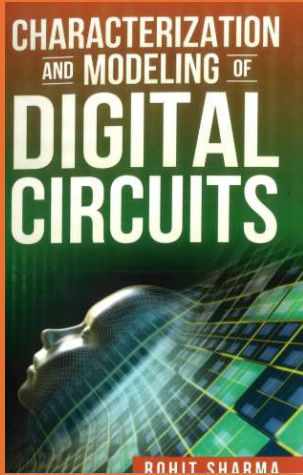




Diseño Circuitos Integrados Digitales

2024 – 2do Cuatrimestre

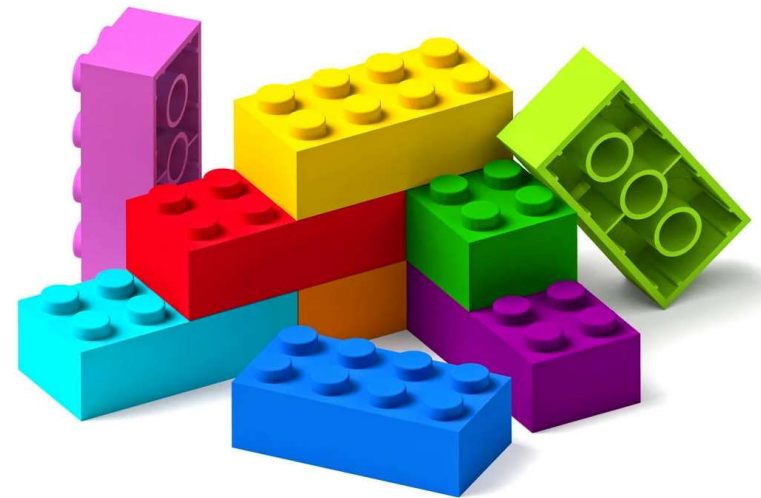
Unidad 6



- Standard Cells: ASIC building blocks
 - Flujo de Diseño
 - Caracterización
 - Qué es y cómo se crea un archivo Liberty?

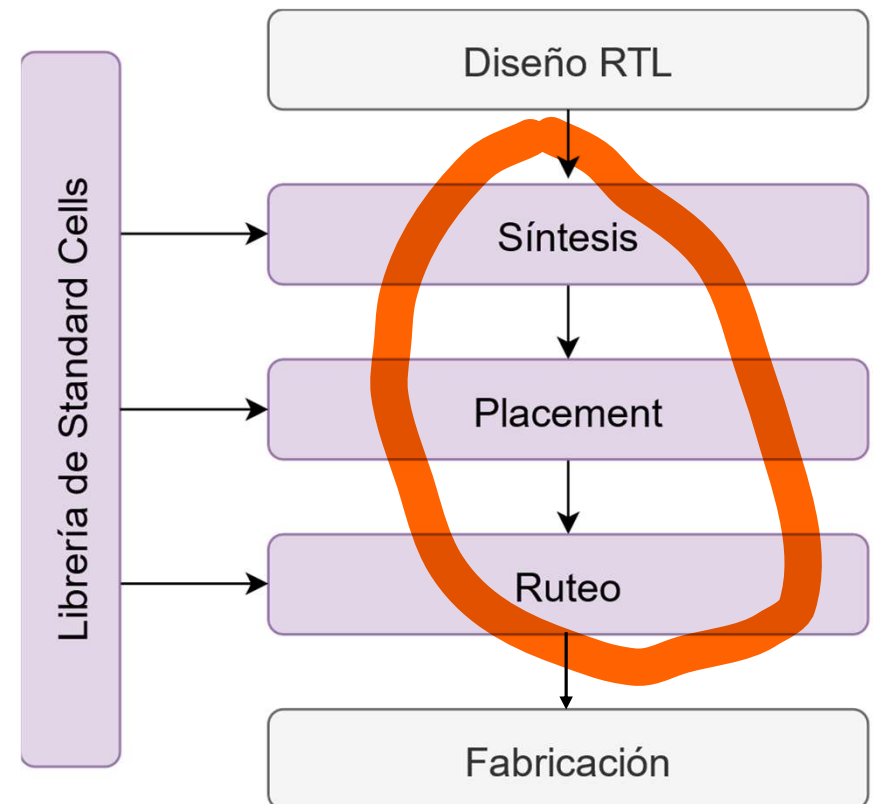
Standard Cells

- Una celda estándar es un grupo de transistores e interconexiones que proporcionan:
 - Función lógica booleana (como AND, OR, XOR, inversores)
 - almacenamiento (flip-flop, latch)
 - Físicas:
 - Antenna para proteger contra acumulación de carga
 - Fillers para llenar espacios vacíos en el diseño,
 - Decoupling capacitors (dcaps) para estabilizar la alimentación de energía.
- Bloque Básico que se combina con otros para formar circuitos más complejos

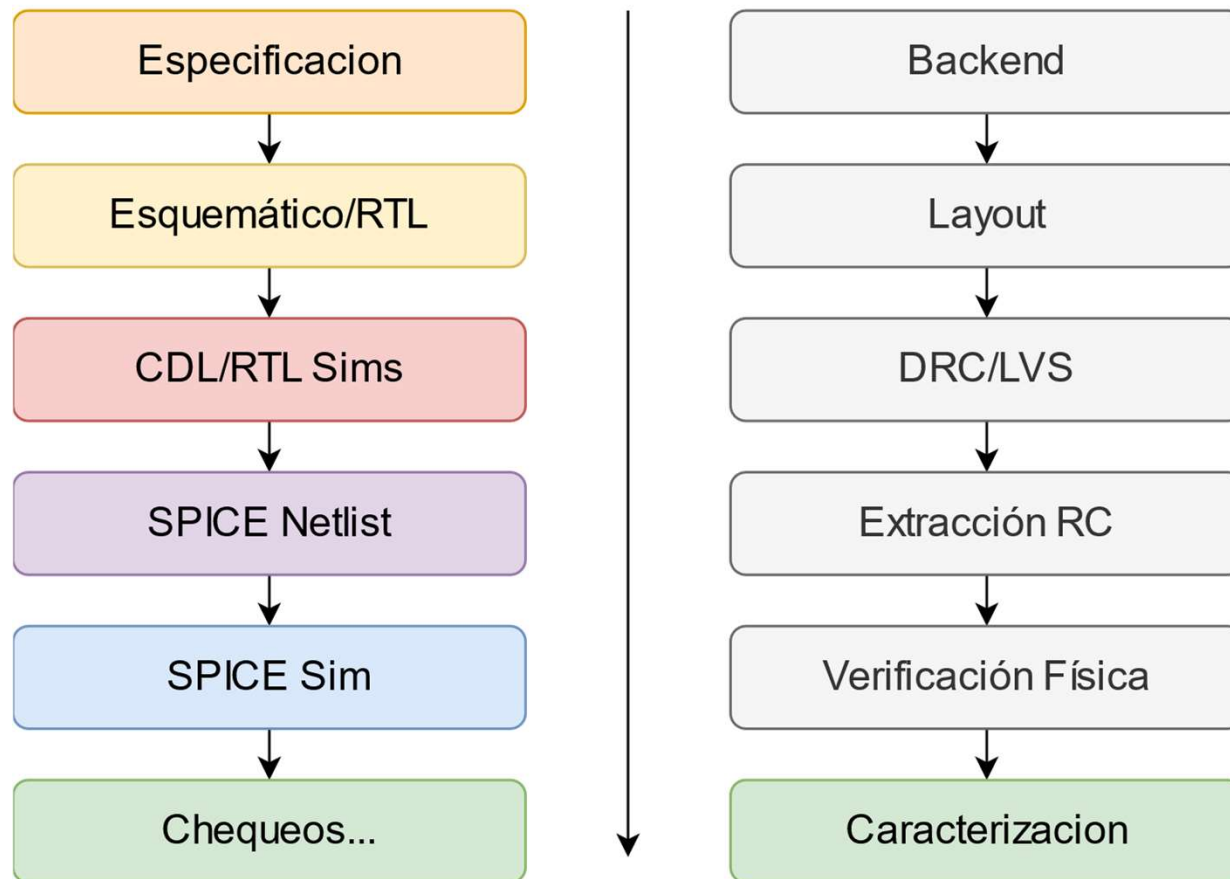


Standard Cells: Dónde se utilizan?

- Conocer la función lógica de una celda no es suficiente para construir circuitos eléctricos funcionales. La velocidad y el consumo de energía de una celda afectan todo el circuito, y estos factores varían según la carga de salida.
- Los modelos de librería de celdas son indispensables para el diseño de chips digitales, desde la simulación lógica hasta la síntesis y ruteo.



Standard Cells: Flujo de Diseño



Standard Cells: Tipos

Compuertas Lógicas
Básicas

HA / FA

Mux

Buffers

ICG

Tie-Cells

AOI/OAI

Flip Flops

Scan Flops

Latches

Fillers

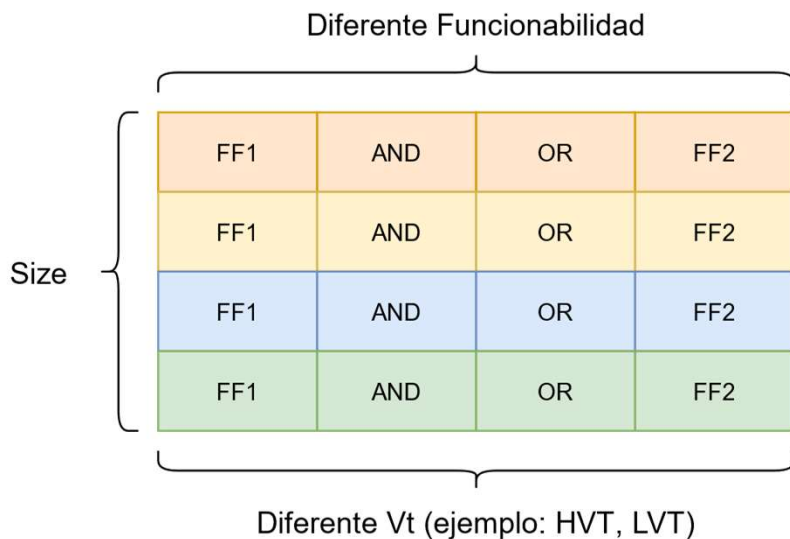
Tap Cells

End-Cap-Cells

D-Cap Cells

Antenna Cells

Standard Cells: Variaciones



- Track Cell

- 6T
- 7.5T
- 9T
- 10.5T
- 12.5T

- Drive Strength

- 1X
- 2X
- 4X
- 8X
- 12X

Width Transistor

- Threshold Voltage:

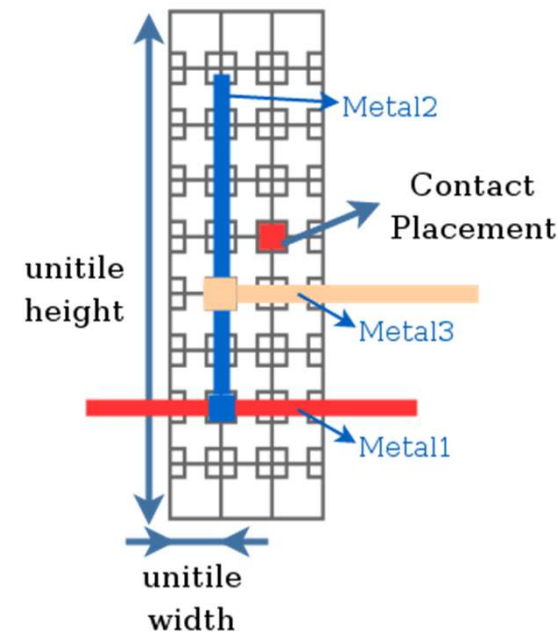
- Ultra Low-VT
- Low-VT
- Standard-VT
- High-VT
- Ultra High-VT

↑ Tox, delay

↓ Leakage Power

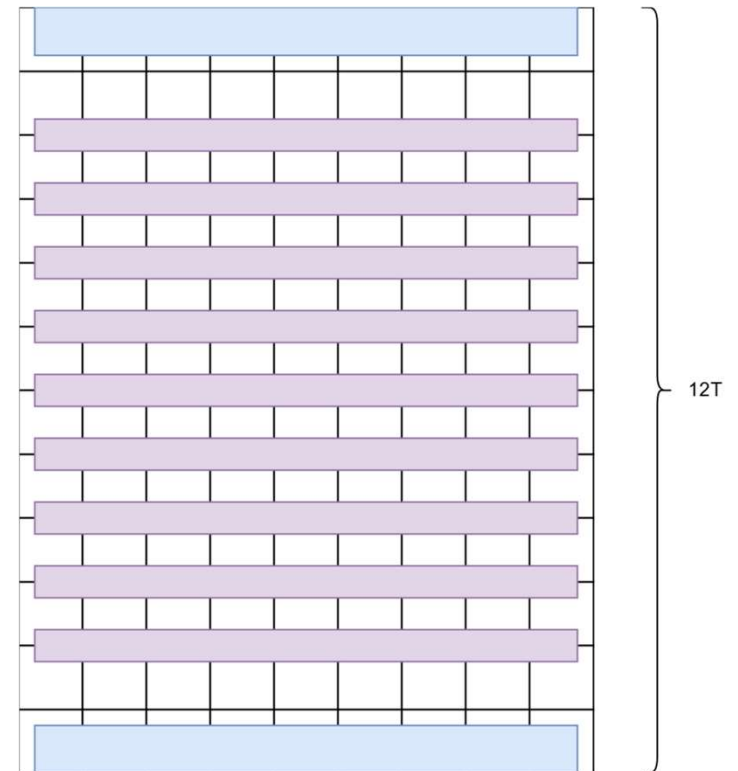
Standard Cells: Tracks

- Una decisión importante de diseño para toda librería es la altura de las celdas
 - Celdas mas altas tienen transistors con mayor width, resultando en mayor velocidad
 - Además, también tienen más espacio disponible para rutear wires, lo que significa una menor congestión.
- La altura de la celda se mide en términos de los “tracks de ruteo” de la capa de metal más baja que no es de interconexión.
 - Un track es el espacio más pequeño por donde podemos rutear un wire.
 - Típicamente se define en términos de las reglas de diseño del PDK
- Generalmente 9-tracks (9T) es el standard
 - 12T se utiliza para diseños de alta velocidad
 - 7T para diseños de alta densidad.



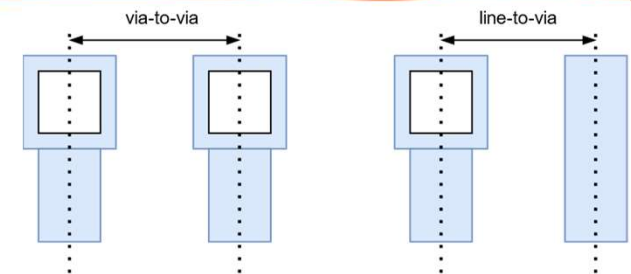
Standard Cells: Tracks (2)

- Ejemplo: 12T
 - Notar que a pesar de tener 12 tracks de alto, podemos utilizar 9 tracks para wires de ruteo
 - Los Power Rails también ocupan lugar dentro de los tracks
- Celdas de distinto track proven una simple solución al tradeoff speed vs area
 - Generalmente no se mezclan en el mismo diseño.
 - Si podriamos tener distintos power domains con distintas librerías.

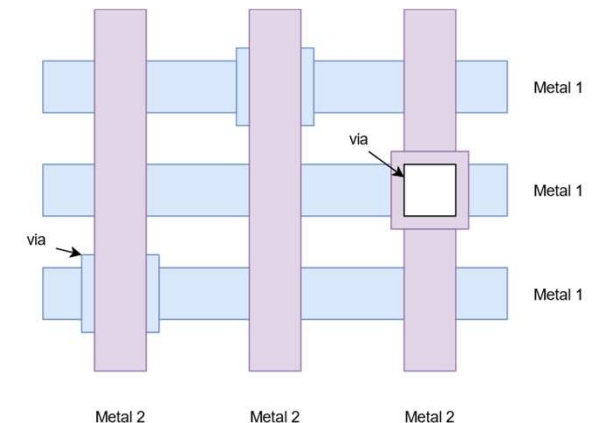


Standard Cells: Track Pitch and Direction

- El paso (***pitch***) con el que se espacian los *wires* definen el tamaño de un ***routing track***.
 - Puede definirse de vía a vía o de línea a vía.
- Para simplificar ***Place and Route***, los *metal layers* se usan en direcciones alternas:
 - Layers impares para ruteo horizontal.
 - Layers pares para ruteo vertical (HVH).
 - En nodos tecnológicos avanzados, es común que los primeros 2 layers de metales se usen para ruteo horizontal.

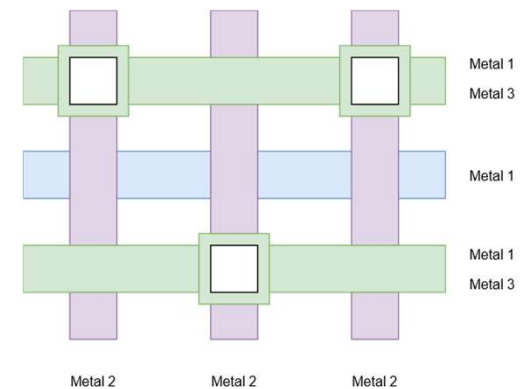


Line to via: Mínimo pitch en donde se puede poner una via al lado de un routing track

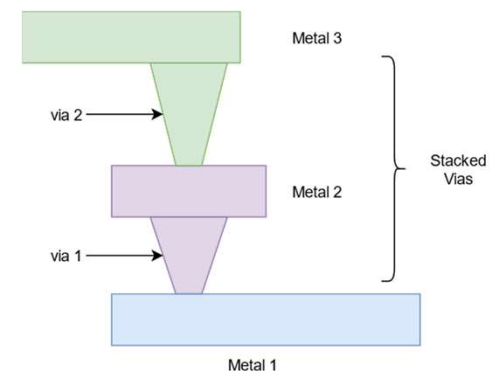


Routing Grid: Definiciones

- Distintos **Layers** con la misma dirección de ruteo pueden tener diferentes grosores y espaciados, lo que crea un **pitch diferente**.
- Ejemplo: Si las pistas de M3 y M1 tienen una proporción de pitch como 11:8, las pistas horizontales se superpondrán, dificultando o incluso haciendo imposible la creación de un contacto entre M3 y M1.
- Es importante usar proporciones simples como 1:1, 1:2, 2:3, 3:4 entre pistas adyacentes con la misma dirección de ruteo.
- Es deseable que las pistas estén alineadas para permitir el uso de vías apiladas.

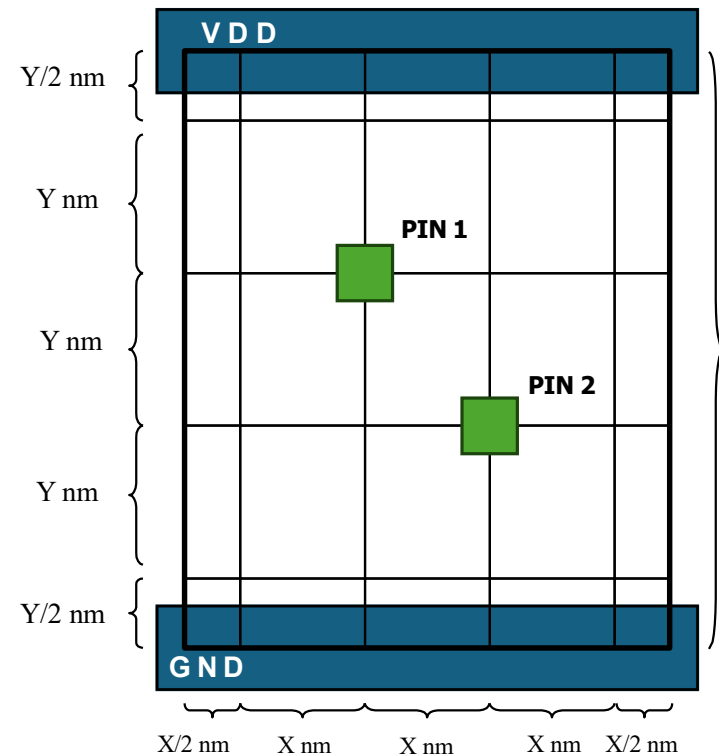


Ejemplo 1:2 M3-M1 ratio



Layout y Power Grid

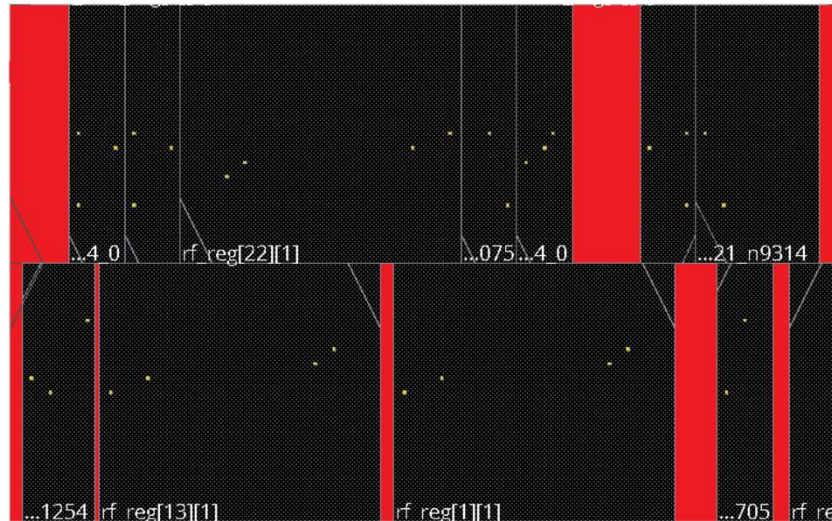
- Los pines deben colocarse sobre la grilla de ruteo para reducir el esfuerzo de routing.
- Pines escalonados (staggered): Colocar los pines de manera escalonada, cuando sea posible (alcanzables por layers de metal horizontales y verticales)
- Power routing con **abutment**: Las señales de power (VDD, GND) se rutean a través de la unión de las celdas estándar, lo que incrementa la densidad.



Power rails diseñado para permitir abutment.

Standard Cells: Dimensiones

- La altura de las celdas estándar debe ser la misma en toda la librería.
- No se permiten espacios entre las celdas y deben llenarse con celdas de relleno (**filler cells**).
- El ancho de todas las celdas estándar debe ser múltiplo entero de la celda filler más pequeña.
 - Cuanto menor sea este divisor común, menos espacio se desperdicia.



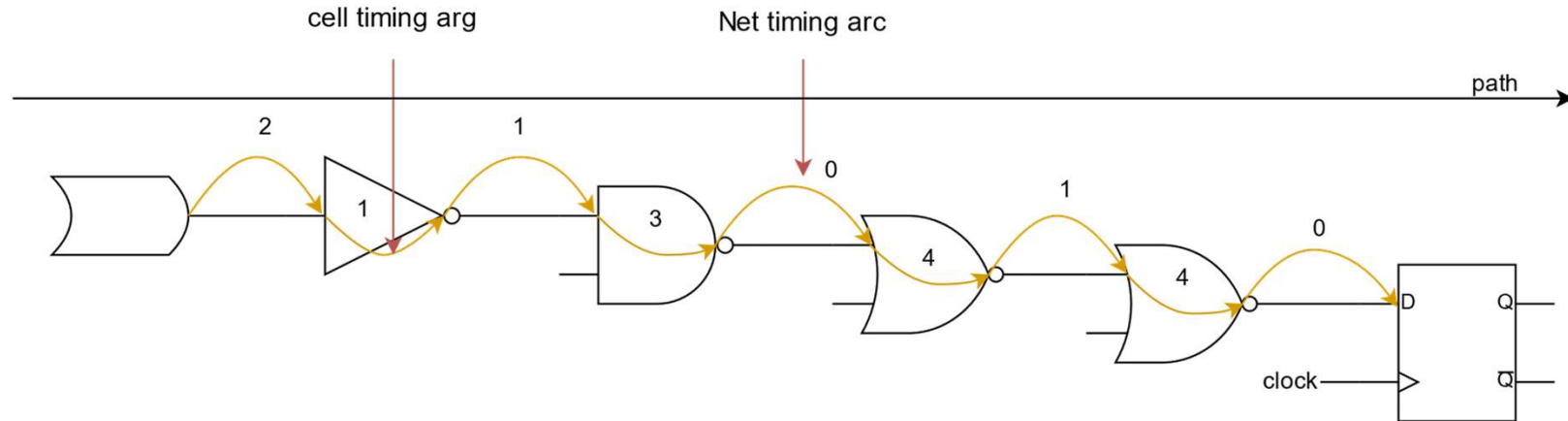
Vista LEF

- LEF/DEF es un formato globalmente utilizado para representar reglas de diseño, información abstracta de celdas estándar y **layout** de circuitos.
- Es un formato completamente basado en texto y utilizado por la mayoría de las herramientas EDA.
- **LEF (Library Exchange Format)** proporciona información abstracta sobre las celdas para ser usada durante **PnR**, como ubicaciones de pines, obstrucciones de routing y restricciones de antena.
- Se generan vistas LEF de todas las celdas para facilitar su uso.

```
MACRO ADDFX1
  CLASS CORE ;
  ORIGIN 0 0 ;
  FOREIGN ADDFX1 0 0 ;
  SIZE 7.04 BY 6.66 ;
  SYMMETRY X Y ;
  SITE 18T ;
  PIN A
    DIRECTION INPUT ;
    USE SIGNAL ;
    PORT
      LAYER met1 ;
      RECT 5.01 1.735 5.3 1.965 ;
      RECT 0.34 1.765 5.3 1.935 ;
      RECT 2.35 1.735 2.64 1.965 ;
      RECT 0.34 1.735 0.63 1.965 ;
    END
  END A
  PIN B
    DIRECTION INPUT ;
    USE SIGNAL ;
    PORT
      LAYER met1 ;
      RECT 4.12 2.475 4.41 2.705 ;
      RECT 0.34 2.51 4.41 2.675 ;
      RECT 4.06 2.505 4.41 2.675 ;
```

Análisis de Tiempo Estático (STA)

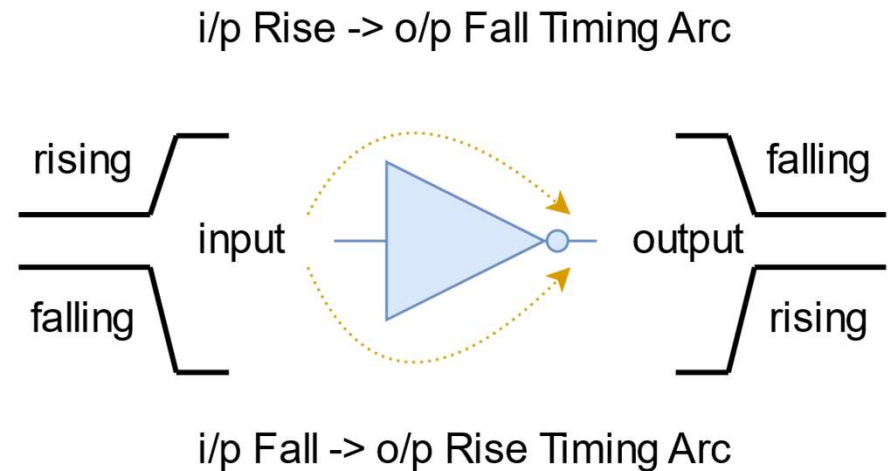
- Método utilizado para verificar **timing** en circuitos digitales.
- Combina cell y net delay para finalmente calcular los **path delay**.
- Los path delays se verifican contra las especificaciones de tiempo.
- Garantiza que el diseño funcione a la velocidad de clock prevista.



$$\text{Path Delay} = 2 + 1 + 1 + 3 + 0 + 4 + 1 + 4 + 0 = 16 \text{ time units}$$

Timing Arc

- Representa una relación causal (un cambio en la entrada causa un cambio en la salida).
- Los **timing arcs** siempre tienen dos pines, ya que la relación es entre estos dos pines.
- Por ejemplo, un inversor no tiene un output timing arc. Tiene un arc que relaciona el pin de salida con el pin de entrada.
- Estos arcs forman los bloques básicos para STA.

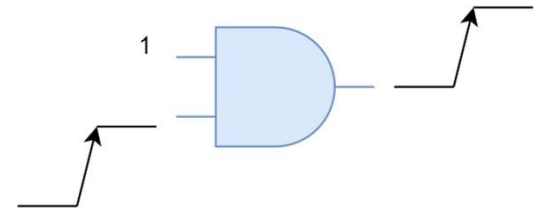


Unateness

Describe cómo cambia la salida de una celda en respuesta a cambios en la entrada.

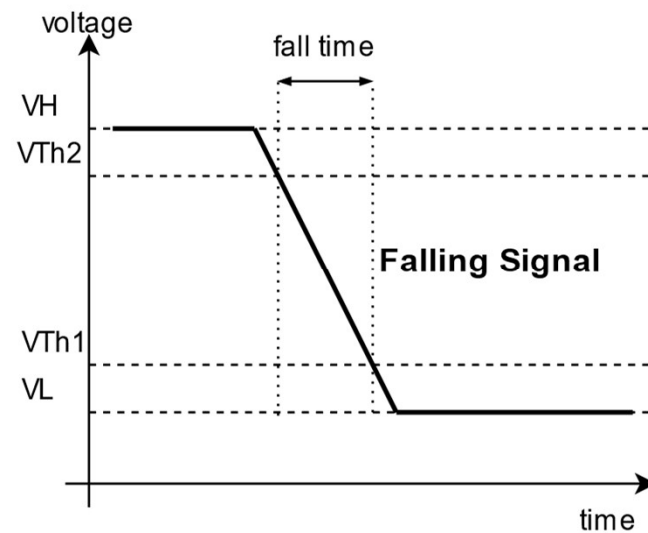
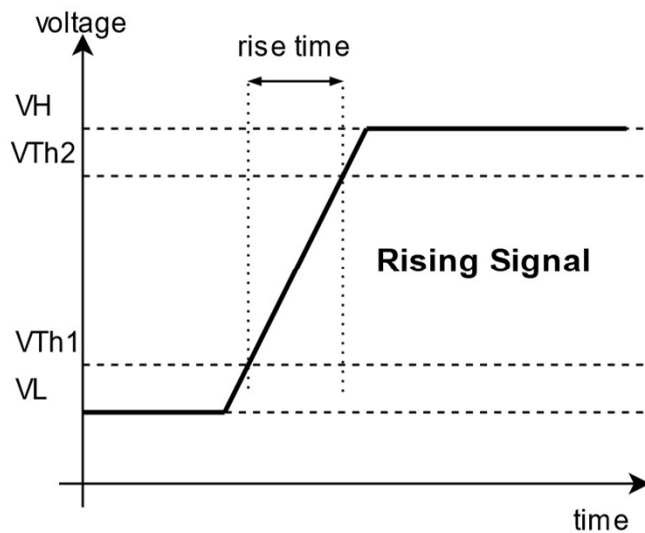
- Existen tres estados de unateness:

- Positive Unate:** Una transición ascendente en la entrada provoca que la salida suba o no cambie, y una transición descendente en la entrada provoca que la salida baje.
Ejemplo: OR y AND
- Negative Unate:** En este caso, una transición ascendente en la entrada provoca que la salida baje o no cambie, y una transición descendente en la entrada provoca que la salida suba.
Ejemplo: NOR y NAND.
- Non Unate:** El estado de otras entradas determina la transición de la salida.
Ejemplo: XOR ya que su salida depende del estado de varias entradas, no de una sola.



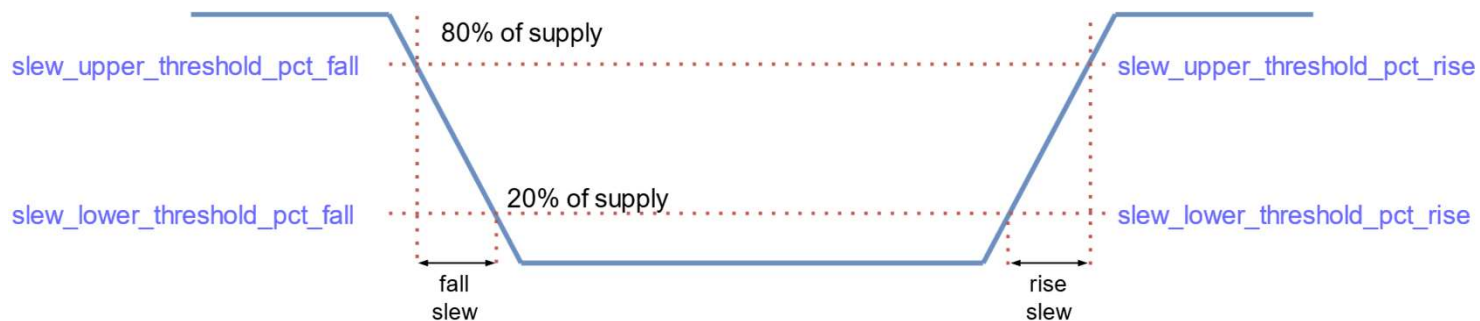
Transition Time

Es el tiempo que tarda una señal en cambiar de estado entre dos niveles específicos. Los tiempos de transición de subida y bajada son propiedades de un timing arc



Slew

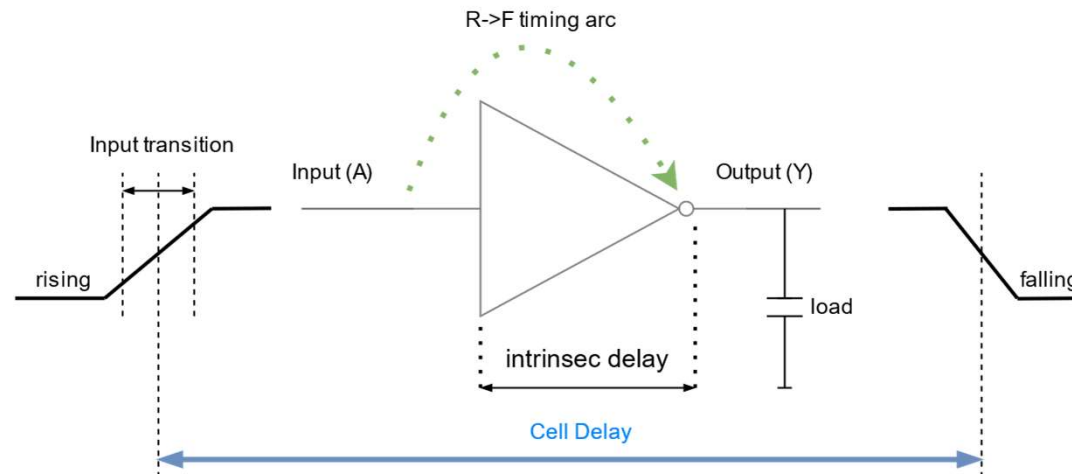
- Se mide típicamente en términos de transition time y está relacionado con el slew rate de la señal.
 - Los umbrales de tiempo de transición de la señal se usan para medir el slew.
- En la figura se especifica que el **falling slew** es la diferencia entre los puntos de tiempo cuando el borde descendente alcanza el 80% y el 20% del valor de suministro.
- **Los umbrales de slew se eligen** típicamente para corresponder a la parte de la forma de onda que es **lineal**.
- En **tecnologías más recientes**, las bibliotecas pueden establecer estos umbrales entre el **30% y el 70%** del valor de suministro.



Cell Delay (propagation delay)

El retardo de propagación a través de una celda se conoce comúnmente como **cell delay**.

- Slew de la señal de entrada y la carga conectada a la celda influyen en el valor.
 - Por lo tanto, los valores de delay se caracterizan por diferentes valores **de input slew y output load**.
- Las mediciones típicas de delay se realizan desde el 50% de la señal de entrada al 50% de la señal de salida.
- El atributo **related_pin** define el/los pin(es) que representan el punto de inicio del timing arc. Este atributo es requerido en todos los grupos de tiempo.



Librerías EDA

- **Formato Liberty (LIB):** Proporciona modelos de temporización con delays, transiciones, tiempos de setup/hold, y función lógica de celdas.
- Formato basado en texto y compatible con herramientas EDA.
- Bibliotecas completamente caracterizadas con scripts para facilitar su expansión.
- Disponibles archivos CCS, ECSM, y VHDL/Verilog con anotaciones.
 - Bloques specify, specparam

Delay Information

Delay(ns) to Y rising :

Cell Name	Timing Arc(Dir)	Delay(ns)		
		First	Mid	Last
AND2X1	A->Y (RR)	0.07489	0.20503	0.65648
	B->Y (RR)	0.07985	0.20601	0.65841
AND2X2	A->Y (RR)	0.08542	0.20967	0.70273
	B->Y (RR)	0.09034	0.20929	0.69907
AND2X4	A->Y (RR)	0.11595	0.24205	0.79319
	B->Y (RR)	0.12084	0.24136	0.78209
AND2X6	A->Y (RR)	0.14558	0.27412	0.86364
	B->Y (RR)	0.15042	0.27399	0.84680
AND2X8	A->Y (RR)	0.17518	0.30612	0.91873
	B->Y (RR)	0.18007	0.30631	0.89836
AND2XL	A->Y (RR)	0.08460	0.22592	0.66372
	B->Y (RR)	0.08984	0.22668	0.66538

Combinational Timing Arcs

- Estos arcs describen el tiempo para elementos combinacionales, que principalmente consisten en el delay.
- El arc estará asociado a un pin de salida, y el related_pin generalmente será una entrada.
- Tipos de combinational timing arcs:

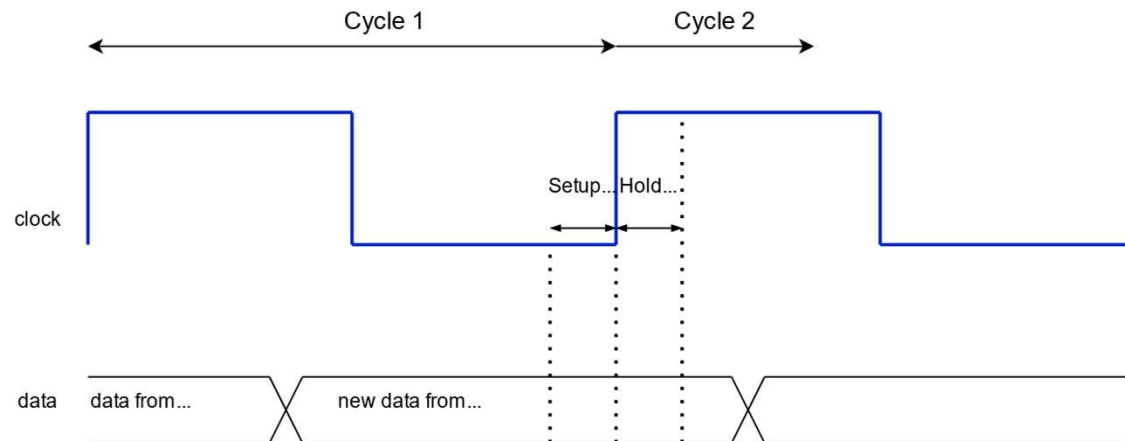
```
combinational  
combinational_rise  
combinational_fall  
three_state_disable  
three_state_disable_rise  
three_state_disable_fall  
three_state_enable  
three_state_enable_rise  
three_state_enable_fall
```

Sequential Timing Arcs

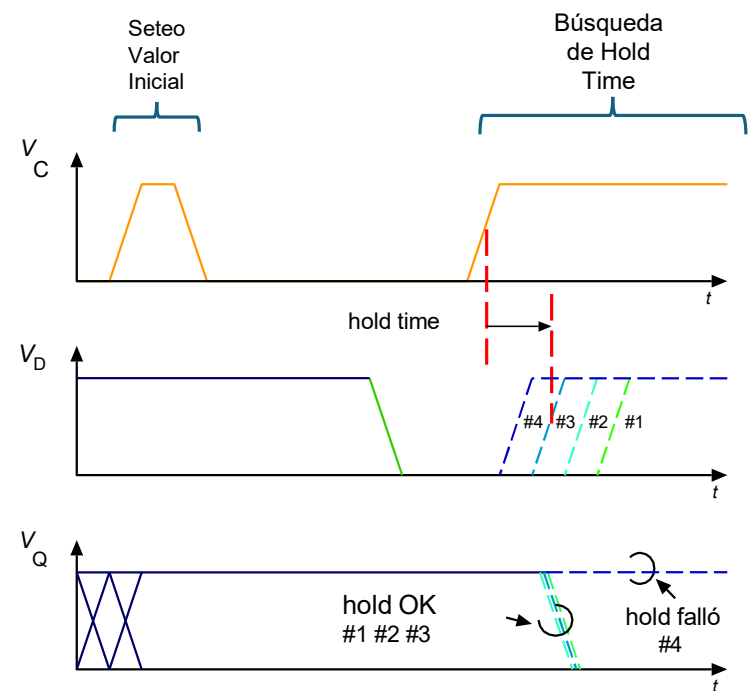
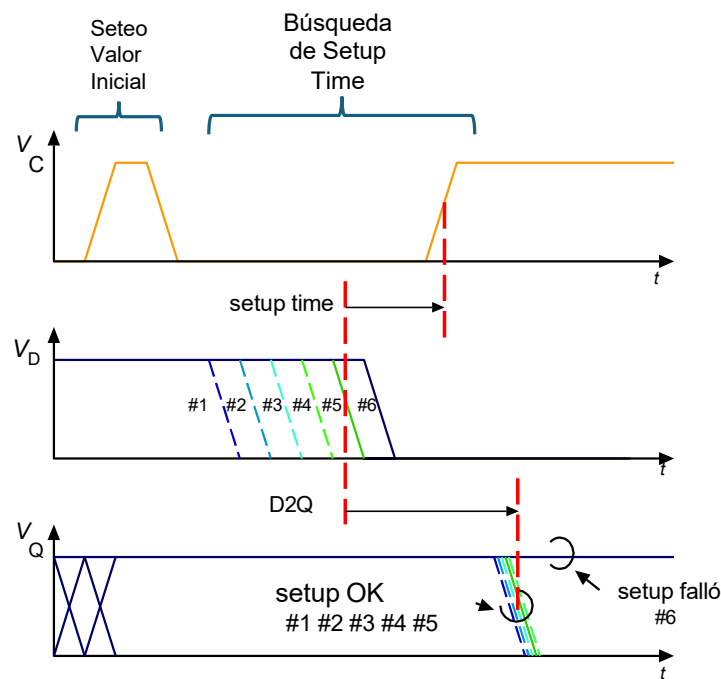
- Estos arcs describen el tiempo para elementos secuenciales.
 - Pueden ser un delay arc (transición de clock a la salida de datos; es decir, de entrada a salida) o un constraint arc (transición de clock a la entrada de datos; es decir, de entrada a entrada).
- Tipos de sequential timing arcs:
 - Edge-sensitive (rising_edge o falling_edge)
 - Preset o clear
 - Setup u hold (setup_rising, setup_falling, hold_rising, o hold_falling)
 - Setup u hold no secuenciales (non_seq_setup_rising, non_seq_setup_falling, non_seq_hold_rising, o non_seq_hold_falling)
 - Recovery o removal (recovery_rising, recovery_falling, removal_rising, o removal_falling)
 - Sin cambio (nochange_high_high, nochange_high_low, nochange_low_high, o nochange_low_low)
- Timing Sense:
 - Este atributo se utiliza en la biblioteca para especificar unateness en el archivo .lib.

Setup and Hold Time

- Estos controles de tiempo sincrónicos aseguran la correcta propagación de datos a través de celdas secuenciales.
- El **setup time** es el tiempo durante el cual los datos de entrada deben permanecer estables antes del borde de disparo del clock.
- El **hold time** es el tiempo durante el cual los datos de entrada deben permanecer estables después del borde de disparo del clock.



Búsqueda Tiempos de Setup/Hold



Setup/Hold: Valores negativos

- **Setup Time Negativo:**

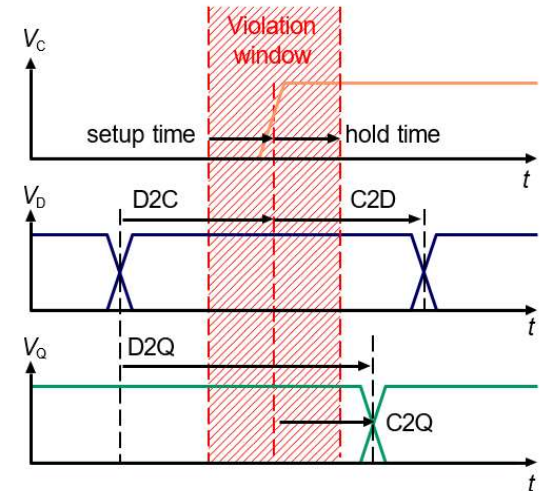
- Los datos pueden cambiar después del borde de clock y aún cumplir con el control de setup.
- Ocurre cuando hay un retraso significativo entre el pin de clock y el uso interno del clock.

- **Hold Time Negativo:**

- Los datos pueden cambiar antes del borde de clock y aún cumplir con el control de hold.
- También ocurre por retrasos entre el pin de clock y el uso interno del clock.

- **Condición Importante:**

- **Si el setup o hold es negativo, su suma debe ser positiva** para que la medición sea válida.
- **Ejemplo:** Un setup time negativo debe estar acompañado de un hold time suficientemente positivo

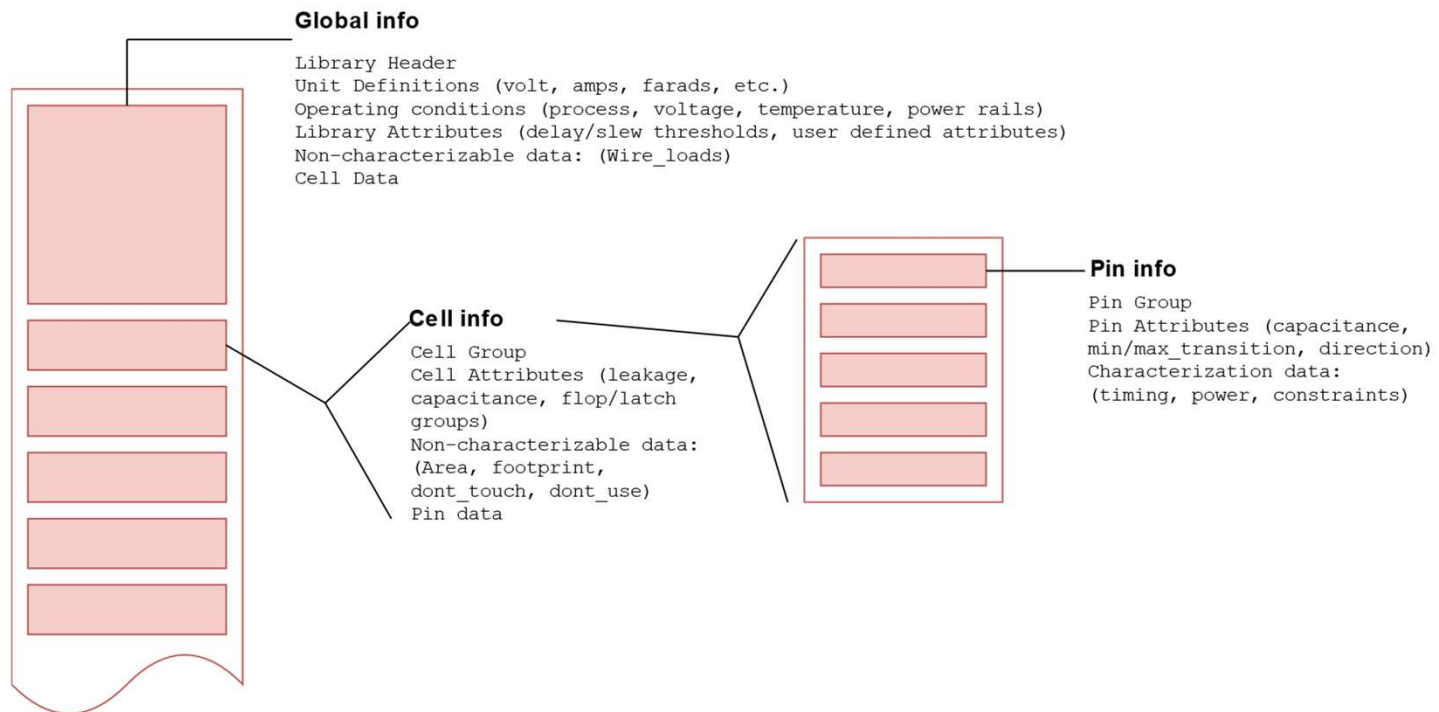


Otras condiciones

- Side Pins Condition, When Condition, SDF_COND, Arcs Dependientes del Estado
- Estos términos se refieren a las expresiones booleanas para un timing arc que especifican el valor (0 o 1) de otros pines en el bloque (conocidos como **side pins**) mientras se prueba un arc.
- El archivo **Liberty** utiliza los atributos "**when**" y "**sdf_cond**" para establecer los valores de tiempo dependientes del estado.

Liberty File: Estructura

- Las partes significativas son:



Unidades

- Las herramientas de STA utilizan las **unidades especificadas** en esta sección para identificar las unidades de medida en la biblioteca.
- No se especifican directamente las unidades para la potencia dinámica interna.
 - Se derivan de la fórmula: corriente * tiempo * voltaje.
 - **Típicamente**, la corriente se mide en mA, el tiempo en ns y el voltaje en volts.
 - Por lo tanto, la potencia dinámica interna se mide en pJ (picojoules).

```
capacitive_load_unit (1,pf);  
current_unit: "1mA"  
leakage_power_unit: "1nW";  
pulling_resistance_unit: "1kohm";  
time_uni: "1ns";  
voltage_unit: "1V"
```

Condiciones Operativas

- El contenido de esta sección es autoexplicativo, excepto por el valor de la variable de proceso.
- El ajuste de la variable de proceso permite que la biblioteca, caracterizada en un punto de proceso particular, se utilice para cálculos en un punto de proceso diferente.
- Este valor generalmente se establece en 1

```
operating_conditions(PVT_09V_125C) {  
    process      : 1;  
    temperature  : 125;  
    voltage      : 0.9;  
}
```

Corner

Worst	Typ	Best
1p5V_190	1p8v_25C	3p2v_n45C

Thresholds

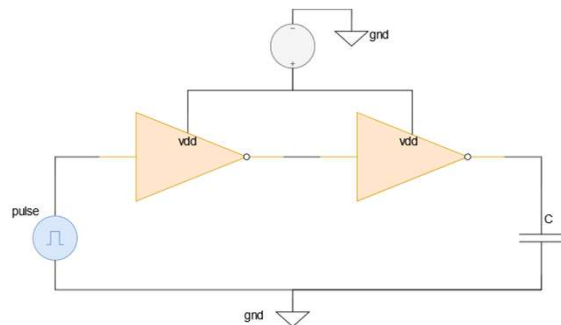
- Los valores de **umbral** de entrada/salida (50% del suministro en el ejemplo) indican los puntos de umbral utilizados para calcular los delays de subida y bajada.
- Los umbrales de slew indican los puntos de corte para los tiempos de transición de subida y bajada de una señal.
- Los valores se especifican como un porcentaje del valor de suministro.

```
slew_lower_threshold_pct_fall : 20.0;  
slew_upper_threshold_pct_fall : 80.0;  
slew_lower_threshold_pct_rise : 20.0;  
slew_upper_threshold_pct_rise : 80.0;  
input_threshold_pct_fall      : 50.0;  
input_threshold_pct_rise      : 50.0;  
output_threshold_pct_fall     : 50.0;  
output_threshold_pct_rise     : 50.0;
```

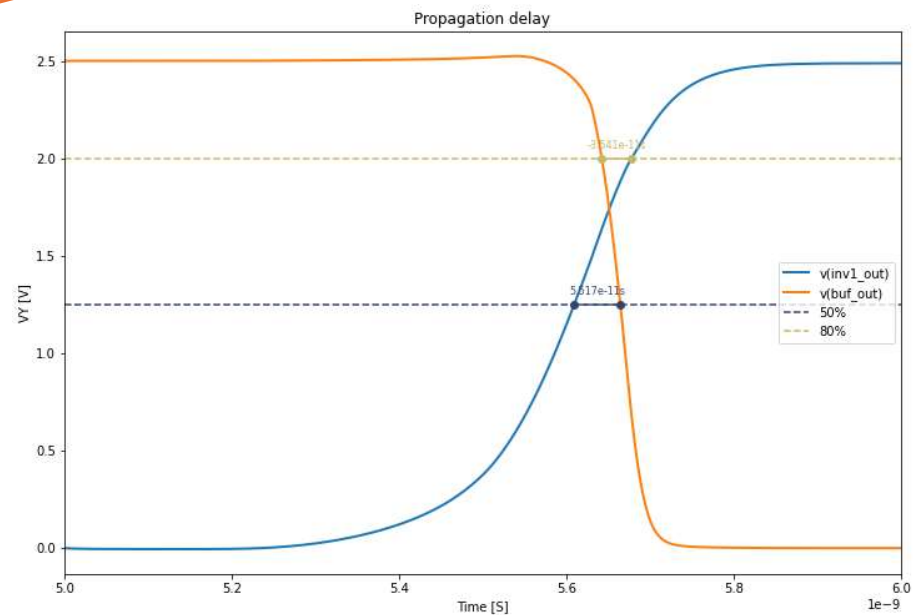
Propagation Delay

```
input_threshold_pct_fall : 50.0;  
input_threshold_pct_rise  : 50.0;  
output_threshold_pct_fall : 50.0;  
output_threshold_pct_rise : 50.0;
```

$Time(out_threshold_pct_*) - Time(in_threshold_pct_*)$



- Al definir el umbral en 50% , obtenemos un propagation delay = 55ps
- Si utilizáramos 80% como umbral, tendríamos -35ps, Tiene sentido esto?



Defaults

Default Cell Attributes:

- `default_leakage_power_density`: Especifica leakage power por unidad de área en una tecnología.
- `default_cell_leakage_power`: Especifica el valor por defecto para celdas sin el atributo `cell_leakage_power`.

Default Pin Attributes:

- `default_output_pin_cap`: Establece un valor por defecto para la capacitancia de todos los pines de salida.
- `default_inout_pin_cap`: Establece un valor por defecto para la capacitancia de todos los pines de E/S.
- `default_input_pin_cap`: Establece un valor por defecto para la capacitancia de todos los pines de entrada.
- `default_fanout_load`: Establece un valor por defecto para la carga de fanout de todos los pines de entrada.
- `default_max_transition`: Establece un valor por defecto para la transición máxima en los pines de salida.
- `default_max_fanout`: Establece un valor por defecto para el fanout máximo en los pines de salida.

Timing and Power

El archivo Liberty generado por Liberate incluye modelos en tablas para especificar información de timing (como delay) de las celdas.

Modelos comunes:

- Non-Linear Delay Model (NLDM)
- Non-Linear Power Model (NLPM)

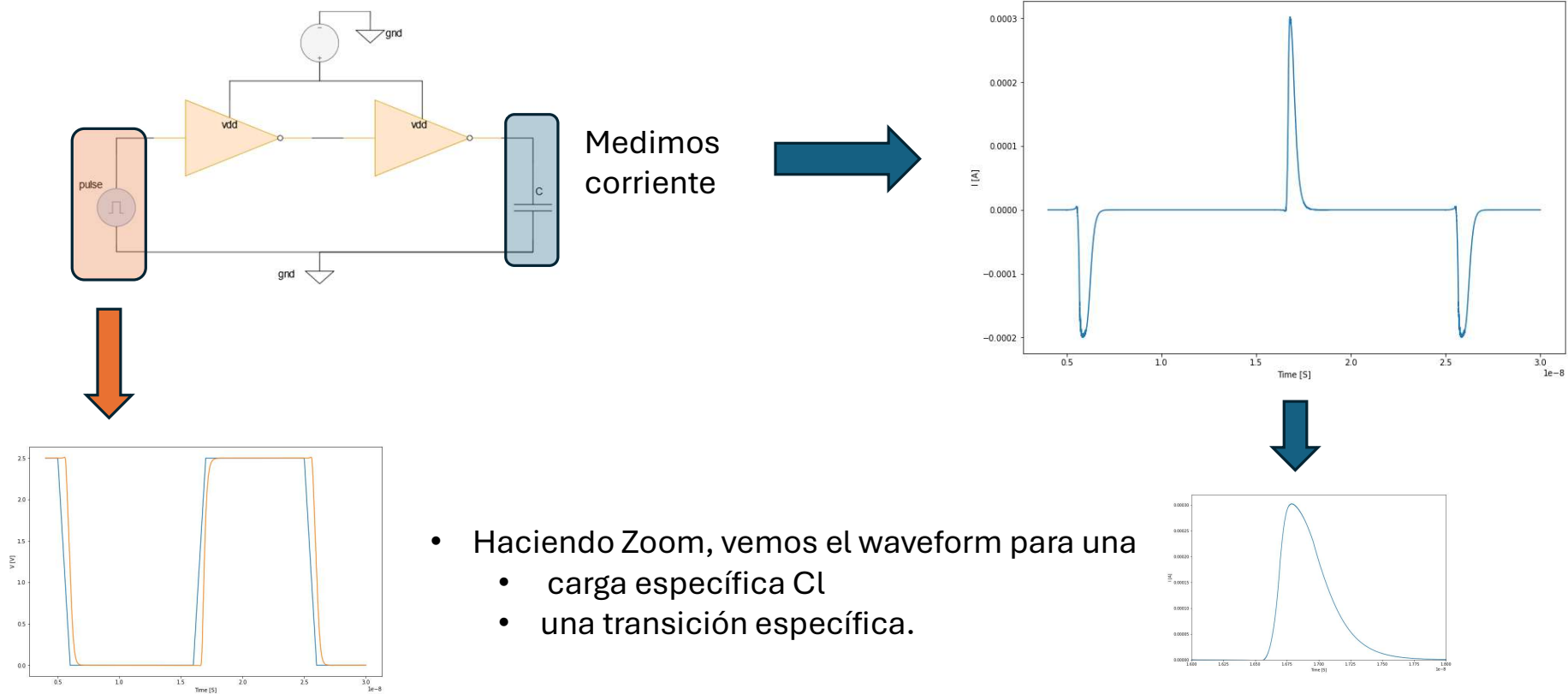
Modelos avanzados:

- Composite Current Source (CCS) y Effective Current Source Model (ECSM) para timing.
- CCSN y ECSMN para ruido.
- CCSP y ECSMP para potencia.

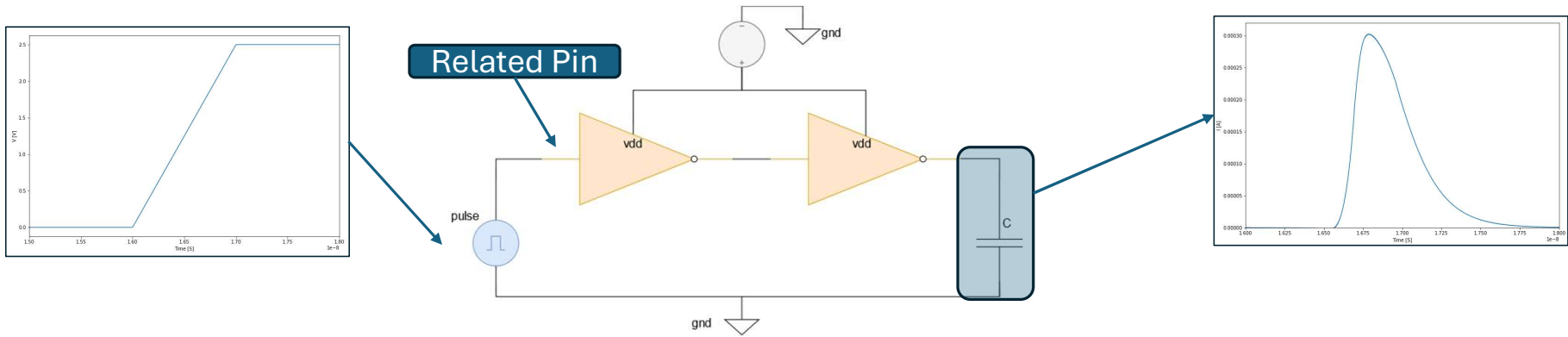
El modelo NLDM representa la información de timing (como delay) en una tabla bidimensional con index_1 y index_2, donde las variables independientes son:

- Input transition time (slew rate).
- Output load capacitance.

Modelos: Output current waveform



Modelos: Output current waveform 2



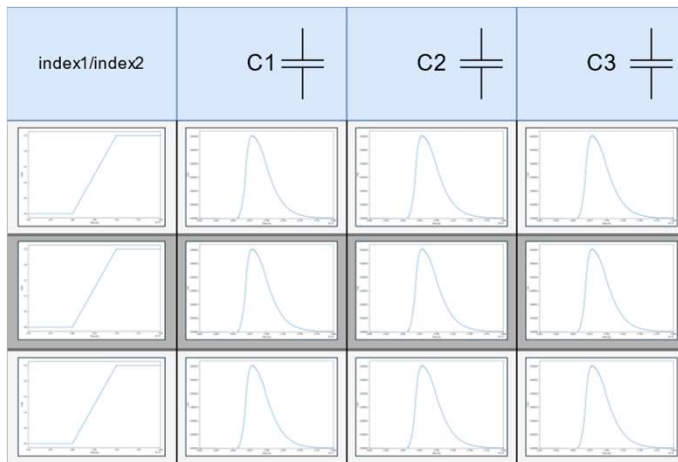
Esto nos permite
construir una Tabla:

index1/index2	C1	C2	C3




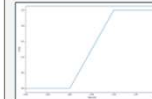
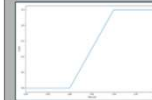

- Composite Current Source (CCS) Format
- De similar manera se realizan output_current_fall

Modelos: Comparativa

- Diferencia entre Non-Linear delay model (NLDM) y Composite Current Source (CCS)



CCS

index1/index2	C1 	C2 	C3 
	Valor Único	Valor Único	Valor Único
	Valor Único	Valor Único	Valor Único
	Valor Único	Valor Único	Valor Único

NLDM

Templates, Lookup Table (LUT)

- Delay LUT Template:

- En la tabla de delay, variable_1 es la transición de la net de entrada (input net transition, index_1) y variable_2 es la capacitancia de la red de salida (output net capacitance, index_2).
- Resultado: Una tabla 3x2 con tres transiciones de entrada y dos valores de carga de salida, lo que da seis valores medidos (3*2).

```
lu_table_template (delay_3x2) {  
    variable_1 : input_net_transition;  
    variable_2 : total_output_net_capacitance;  
    index_1 ("0.01, 0.02, 0.03");  
    index_2 ("0.004, 0.005");  
}
```

Power LUT Template

- Similar a la tabla de delay, también es una matriz de 3x2, resultando en seis mediciones de potencia.

```
power_lut_template (power_3x2) {  
    variable_1 : input_transition_time;  
    variable_2 : total_output_net_capacitance;  
    index_1 ("0.01, 0.02, 0.03");  
    index_2 ("0.004, 0.005");  
}
```

Constraints LUT Template

- Las tablas bidimensionales de restricciones (setup, hold) están basadas en los tiempos de transición en el pin restringido (data) y el pin relacionado (clock).
- **Ejemplo:** Un template 2x2 donde el pin restringido y el pin relacionado pueden ser variable_1 o variable_2. Una vez definidos, estas designaciones son consistentes en todas los templates de la biblioteca.

```
lu_table_template (constraint_2x2) {  
    variable_1 : constrained_pin_transition;  
    variable_2 : related_pin_transition;  
    index_1 ("0.01, 0.02");  
    index_2 ("0.01, 0.02");  
}
```


Cell Model Group

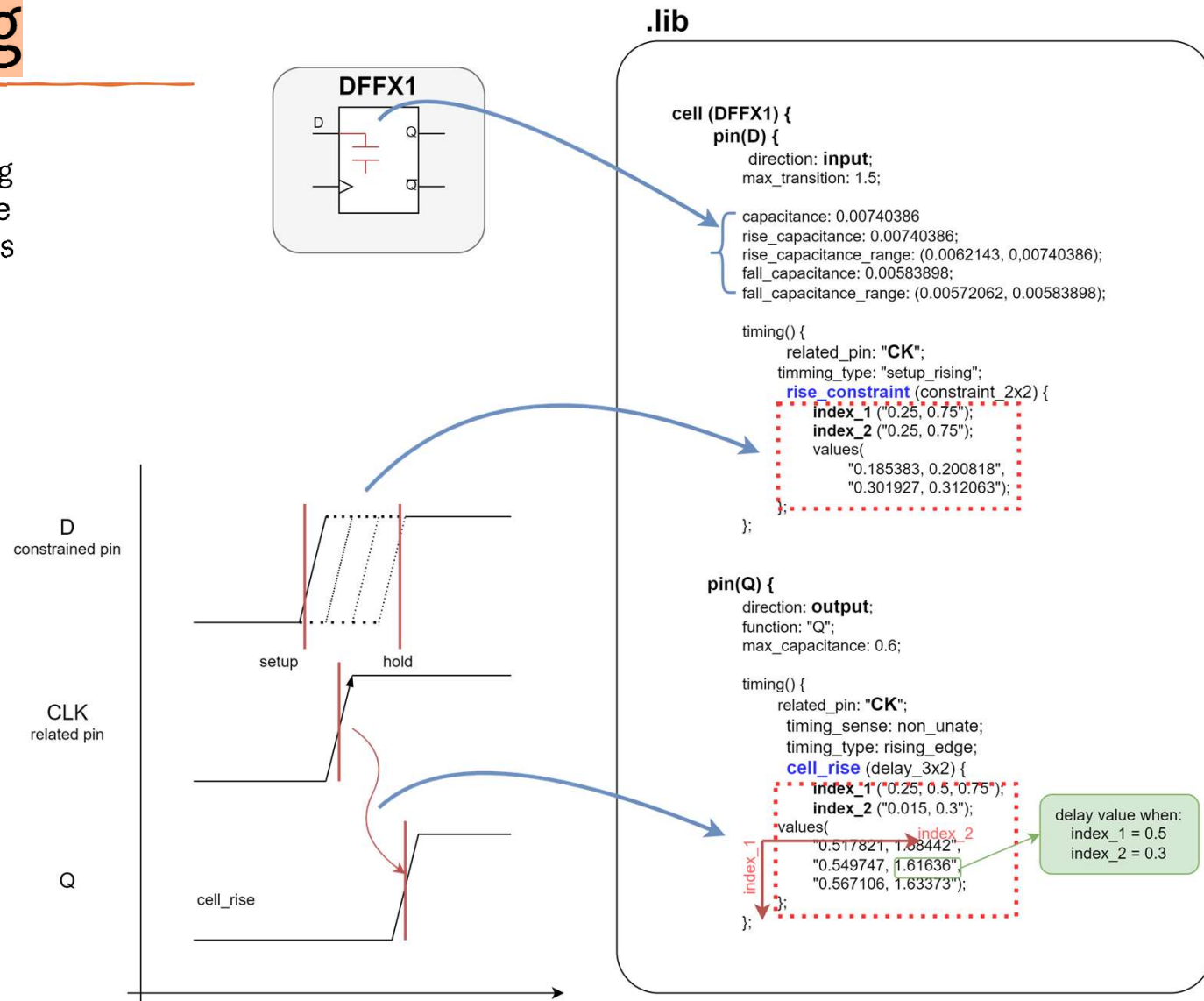
Pin Capacitance

- Normalmente, la capacitancia solo se especifica para los pines de entrada de la celda.
- La capacitancia básica se expresa como un valor único.
- También puede contener valores separados de capacitancia para las transiciones ascendentes y descendentes del pin de entrada.
- Los valores de capacitancia pueden refinarse aún más usando un rango con límites superior e inferior especificados.

```
pin (A) {  
    direction : input;  
    max_transition      : 0.6;  
    capacitance         : 0.00053;  
    rise_capacitance    : 0.00053;  
    rise_capacitance_range : (0.00043, 0.00053);  
    fall_capacitance    : 0.00042;  
    fall_capacitance_range : (0.00041, 0.00042);  
}
```

Cell Timing

- La sección de timing para un pin contiene todos los timing arcs asociados con él.



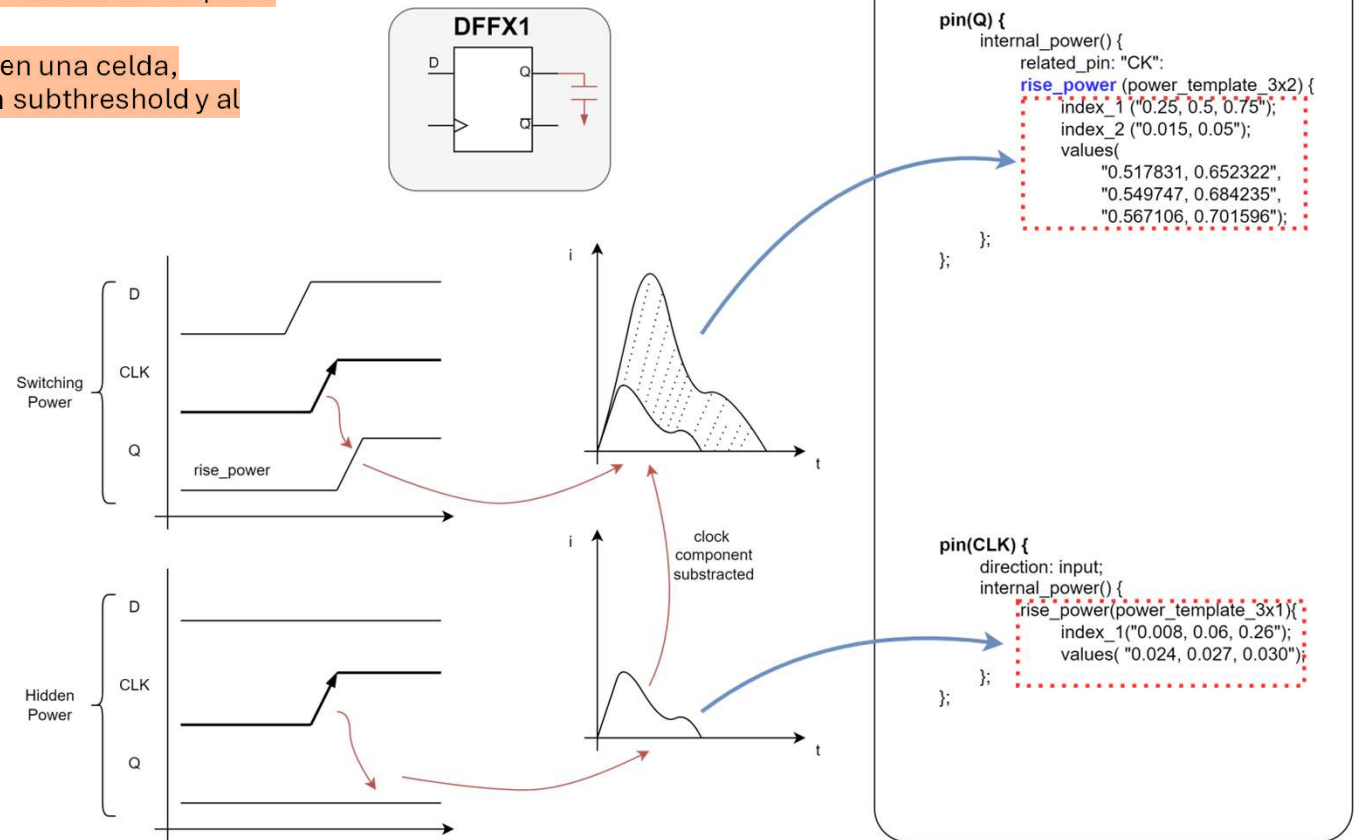
Cell Power

- La información de potencia se especifica en:

- Active power** está relacionado con la actividad en los pines de entrada y salida de la celda.
- Leakage power** es la corriente de fuga en una celda, principalmente debido a la conducción subthreshold y al gate-oxide.

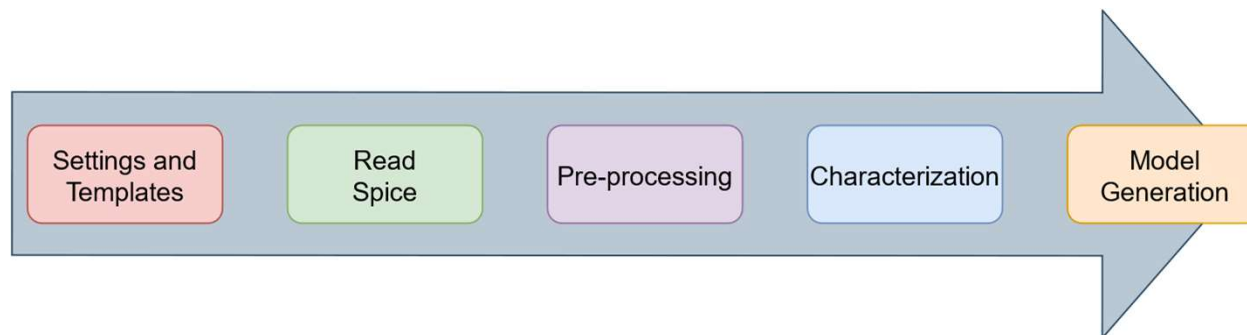
- Active power incluye:

- Output switching power:** Depende de la carga capacitiva y la frecuencia de conmutación.
- Internal switching power:** Consumo de potencia dentro de la celda cuando hay actividad en la entrada/salida.



Liberate: Flujo de Caracterización

- **Settings and Templates:**
 - Se definen celdas, modelos y condiciones PVT para la sesión de caracterización.
- **Read Spice:**
 - El comando read_spice prepara a Liberate para procesar la celda, reconocer su funcionalidad y generar vectores.
- **Pre-processing:**
 - Se procesa el netlist de la celda leído durante read_spice. Se identifican caminos donde los pines y related pins interactúan estáticamente y se asocian con la capacitancia de los pines de entrada. Estos caminos se guardan para simulación posterior.
- **Caracterización:**
 - Se realiza una caracterización precisa usando un simulador SPICE (como Spectre APS) en todos los arcs para todos los slews y cargas.
- **Model Generation:**
 - Después de caracterizar todos los arcs, los resultados se combinan y se escribe un archivo Liberty (.lib) final.



Conclusiones



Titulo

- texto