EECS251B : Advanced Digital Circuits and Systems

Lecture 13 – Delay Models

Borivoje Nikolić

HBCUs Taking Bite of Apple's Chip Engineer Effort.

Feb 27, 2024, EETimes. "What we found is, it's kind of a magical thing, that when you get VLSI (very large-scale integration) engineers from Apple together with students, pretty much every single time you do that, good things happen," Zerbe said.

The alchemy he describes could happen during the times he and his other engineers leave their Cupertino, Calif., Austin, Texas, and other locations to travel to any of the dozen or so schools in the NSI. They include Carnegie Mellon, Stanford and UCLA Berkeley, the inaugural group to become involved in 2019 and 2020. In 2021, Apple added four historically black colleges and universities (HBCUs), including Alabama A&M, Howard, Morgan State and Prairie View A&M.

Jared Zerbe, Apple's director of engineering for hardware technology. (Source: Apple)

Announcements

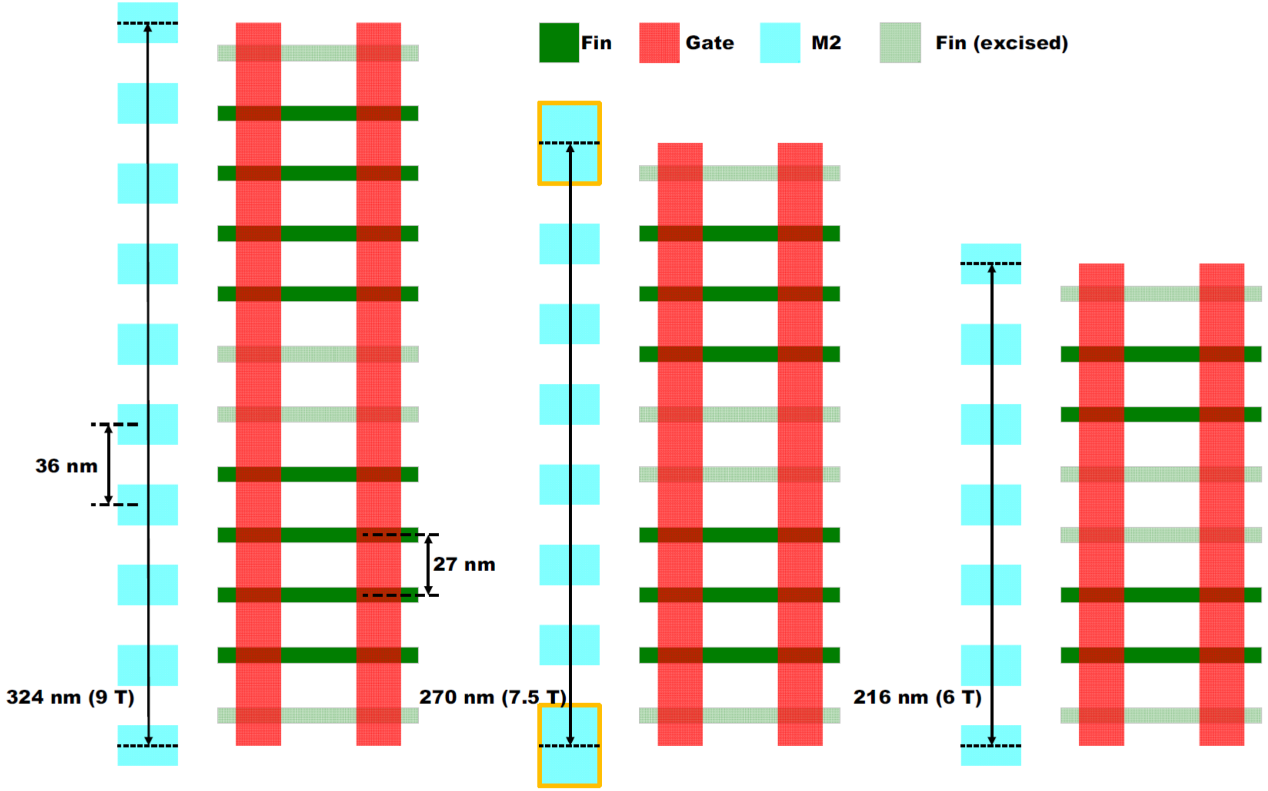
- Lab 5 waiting on PDK correction
- Start project phase 1
  - Spec doc due next week
- Homework 2 due next week
  - Quiz 2 on March 12

Standard Cells in ASAP7

FinFET Standard Cells

ASAP7

- **Standard cell height selection is application specific**
  - **Related to fins/gate, i.e. drive strength**
- **Gear ratio: fin-to-metal pitch ratio**
  - **Cell height needs to be integer # of fins and (mostly) an integer # of metals accessing the cell pins (e.g. M2)**



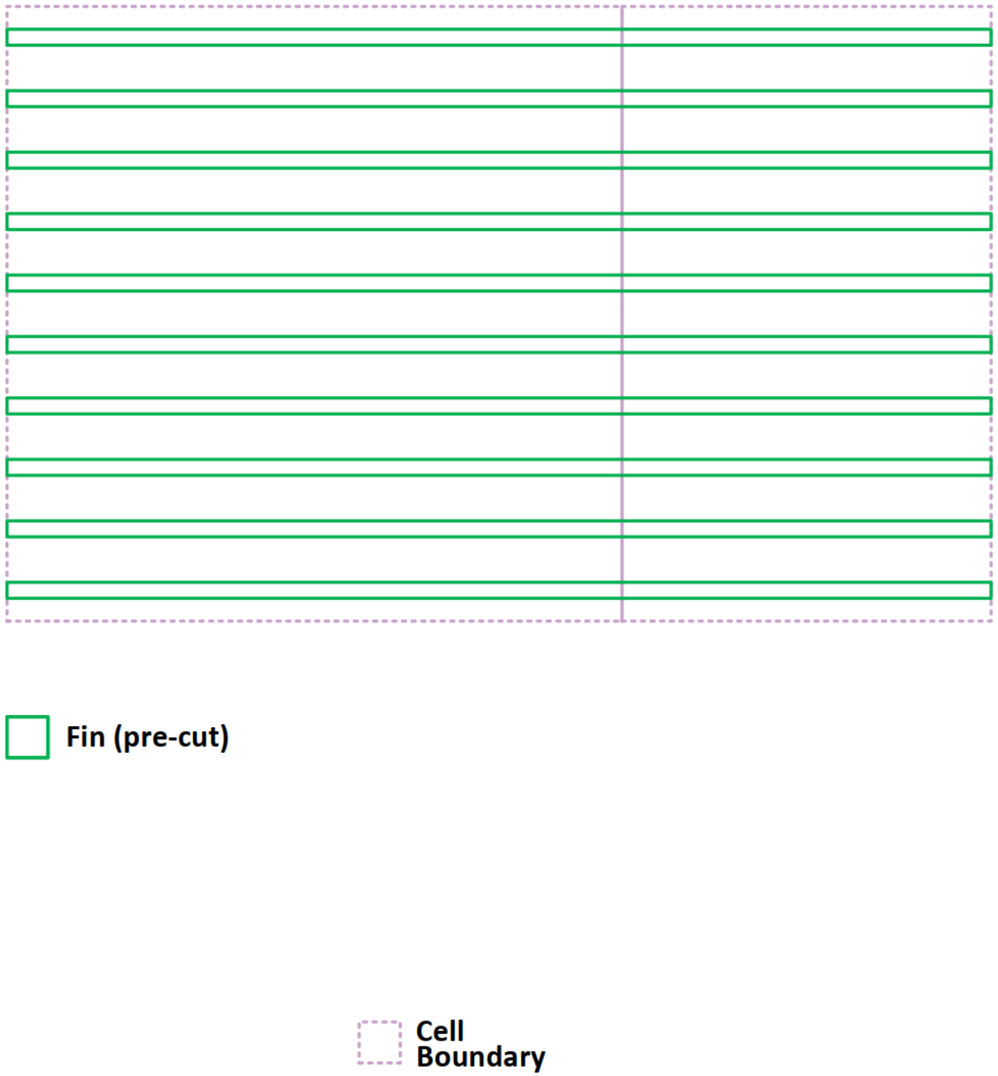
- **12 fin pitches, 9 M2 tracks**
  - **Easy intra-cell routing, rich library**
  - **Wasteful for density**
- **10 fin pitches, 7.5 M2 tracks**
  - **Rich library without overly difficult routing or poor density**
  - **Allows wide M2 power rails**
- **8 fin pitches, 6 M2 tracks**
  - **Difficult intra-cell routing, diminished library richness**
  - **Limited pin access**

ICCAD 2017 Embedded Tutorial ASAP7 16

V. Vashishtha, ICCAD'17

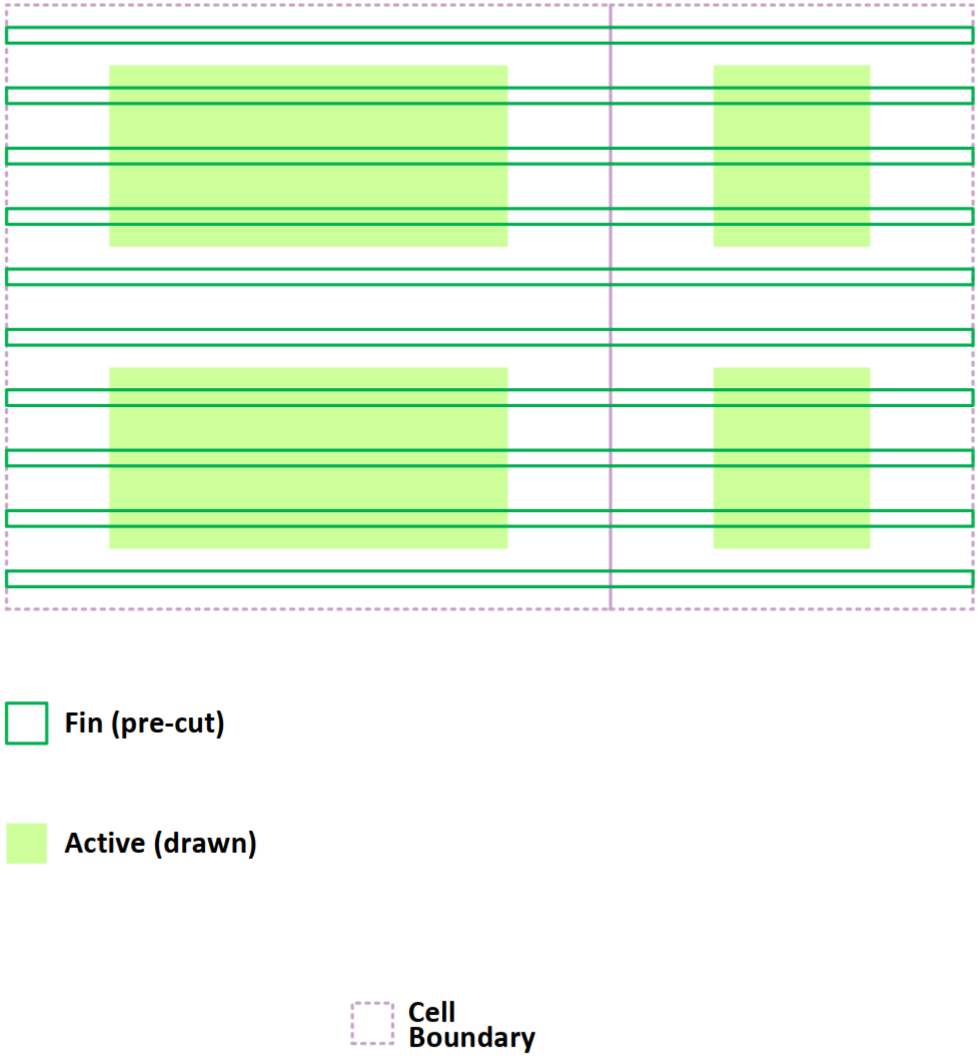
ASAP7 Standard Cells

- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**



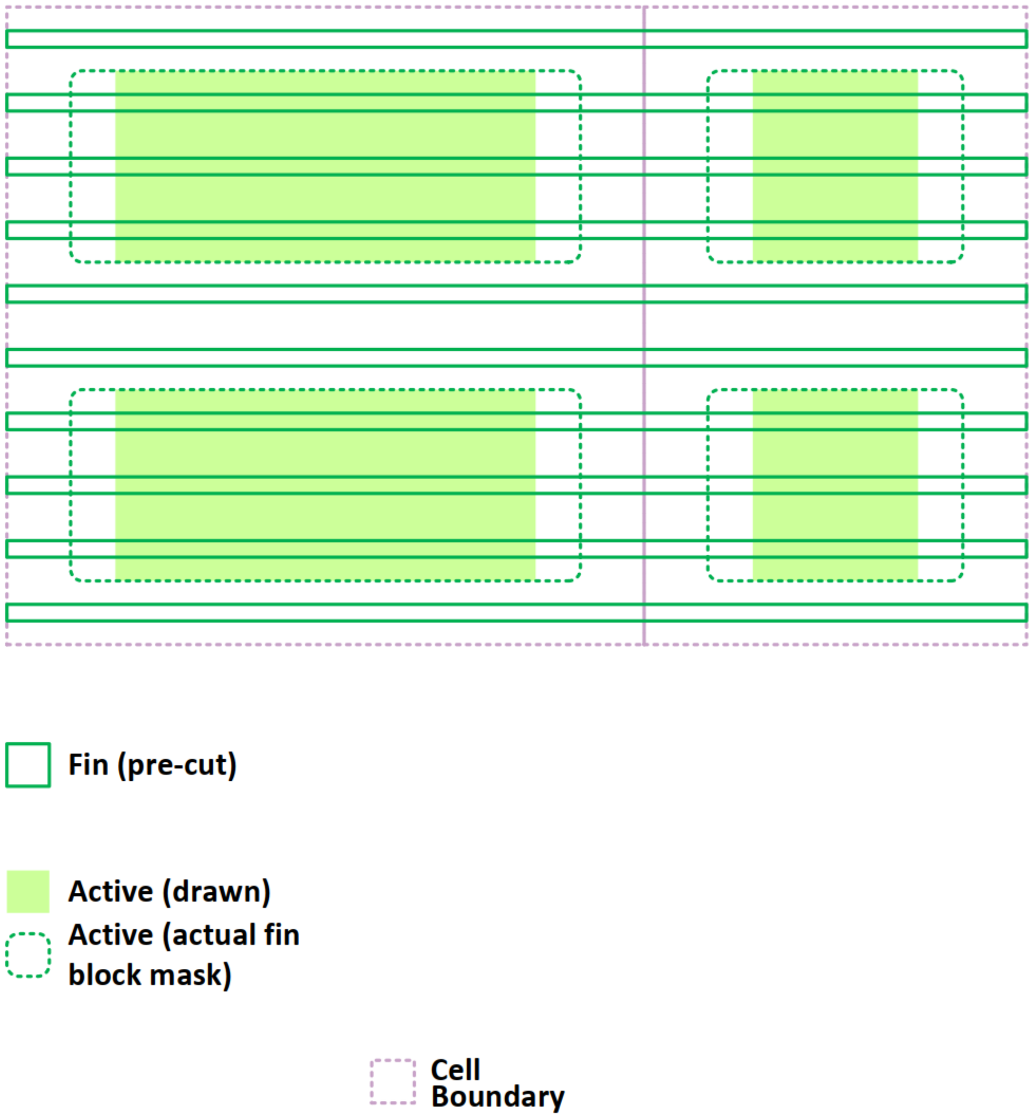
ASAP7 Standard Cells

- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**



ASAP7 Standard Cells

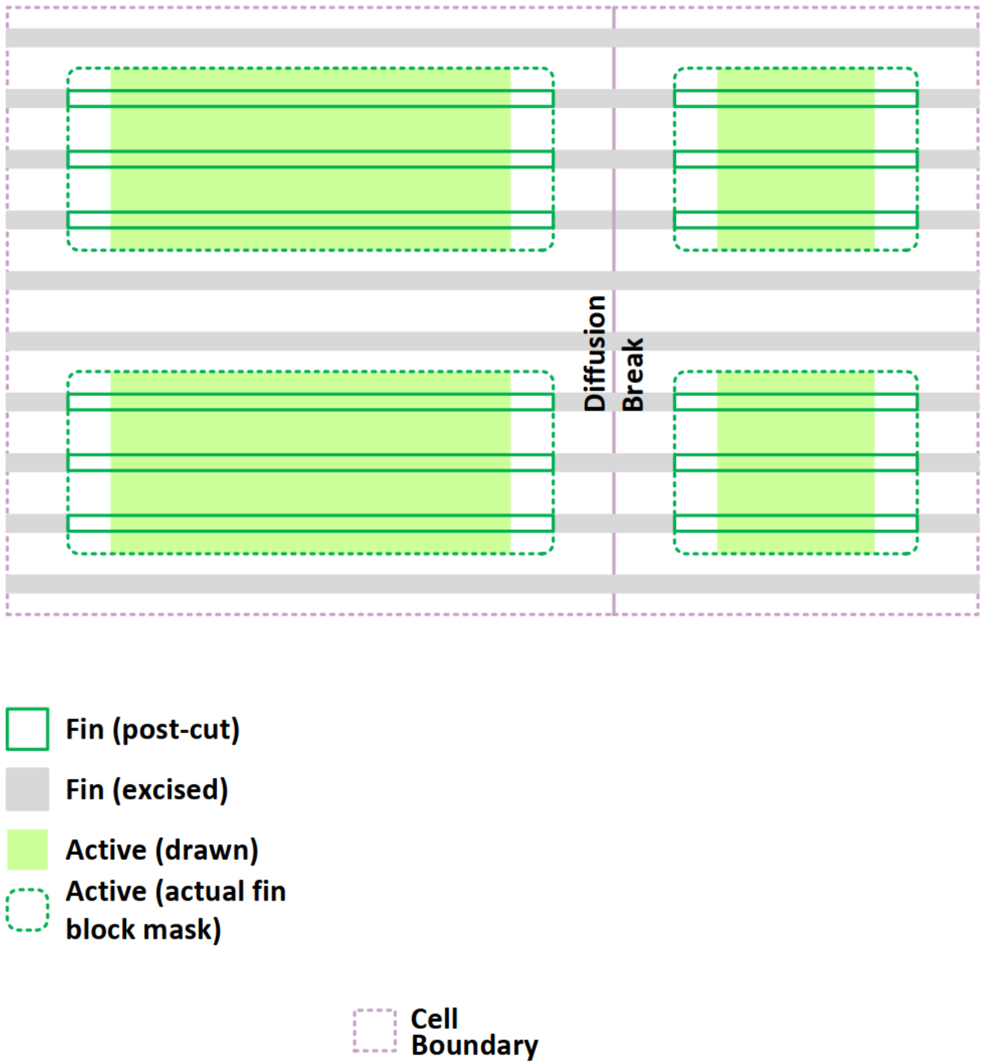
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**



ASAP7 Standard Cells

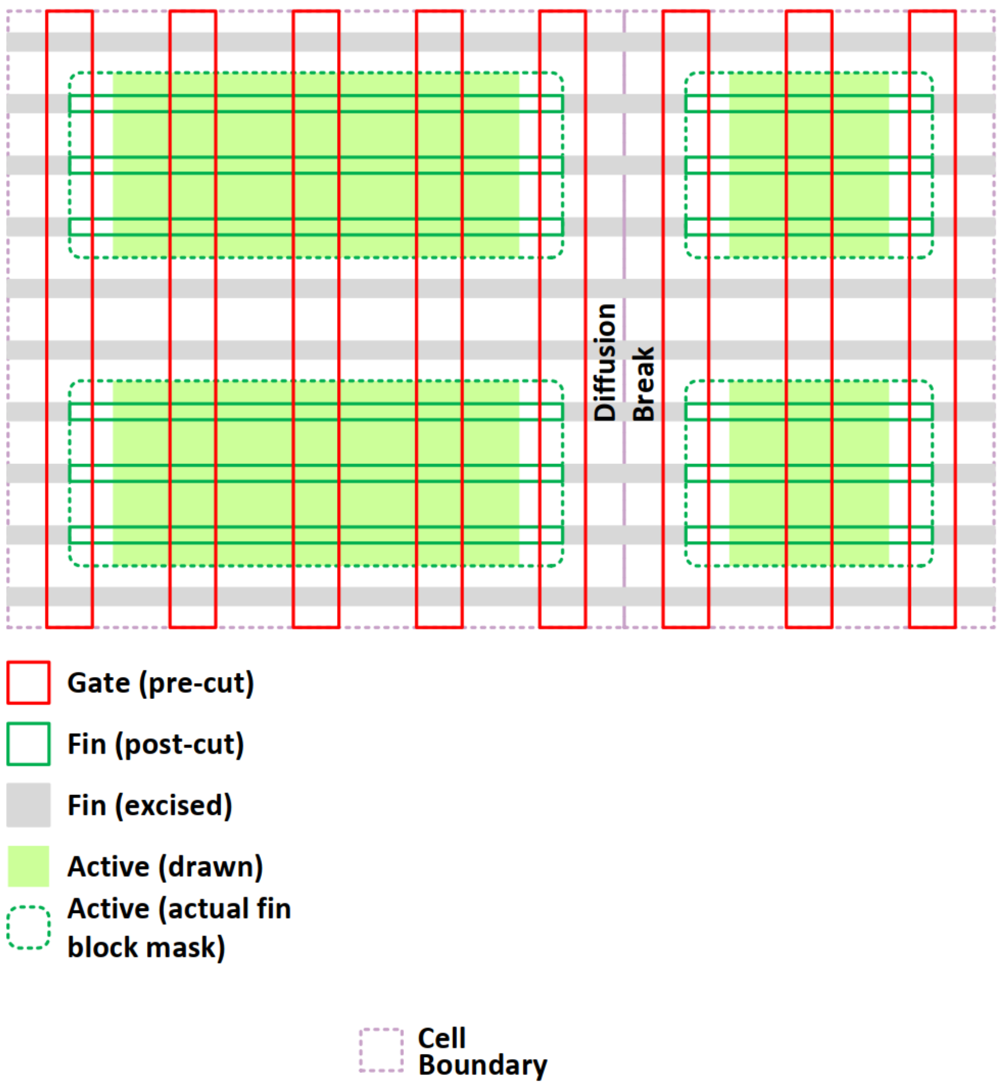
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**

- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

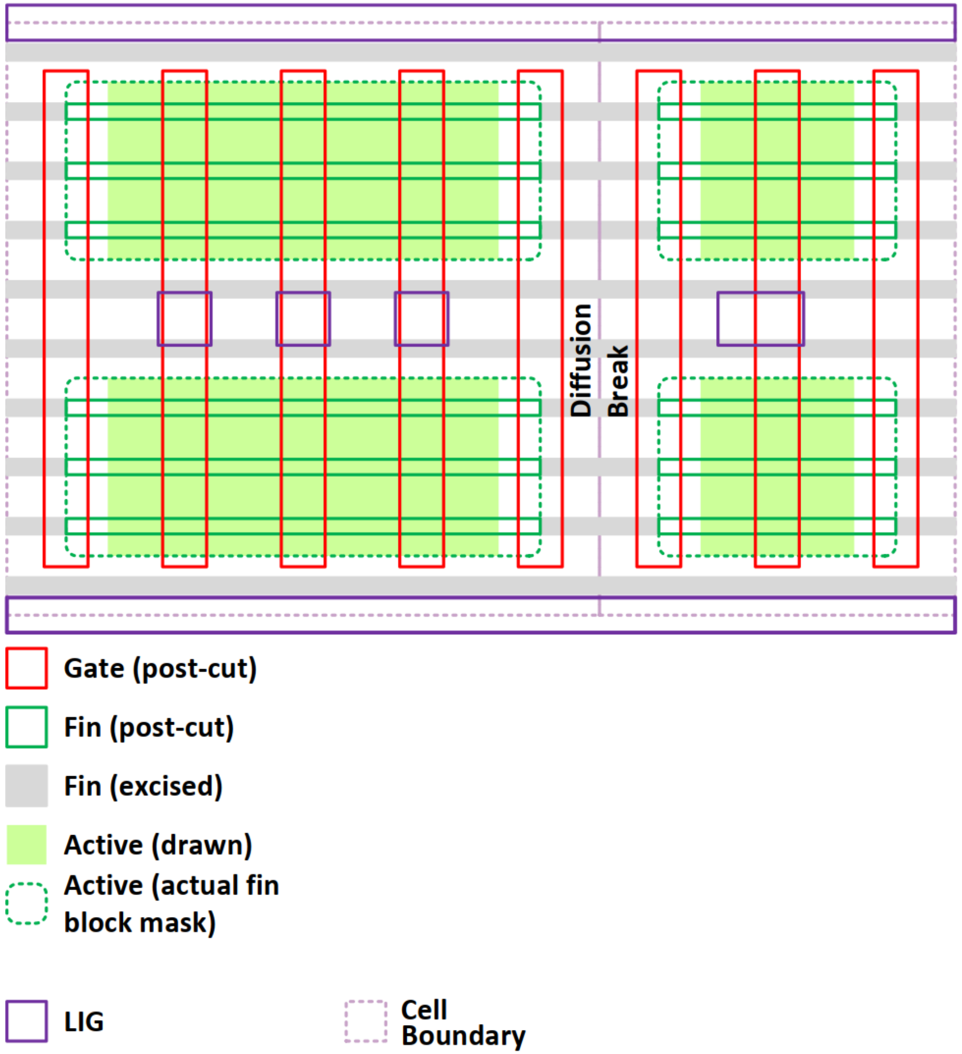
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**
- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**

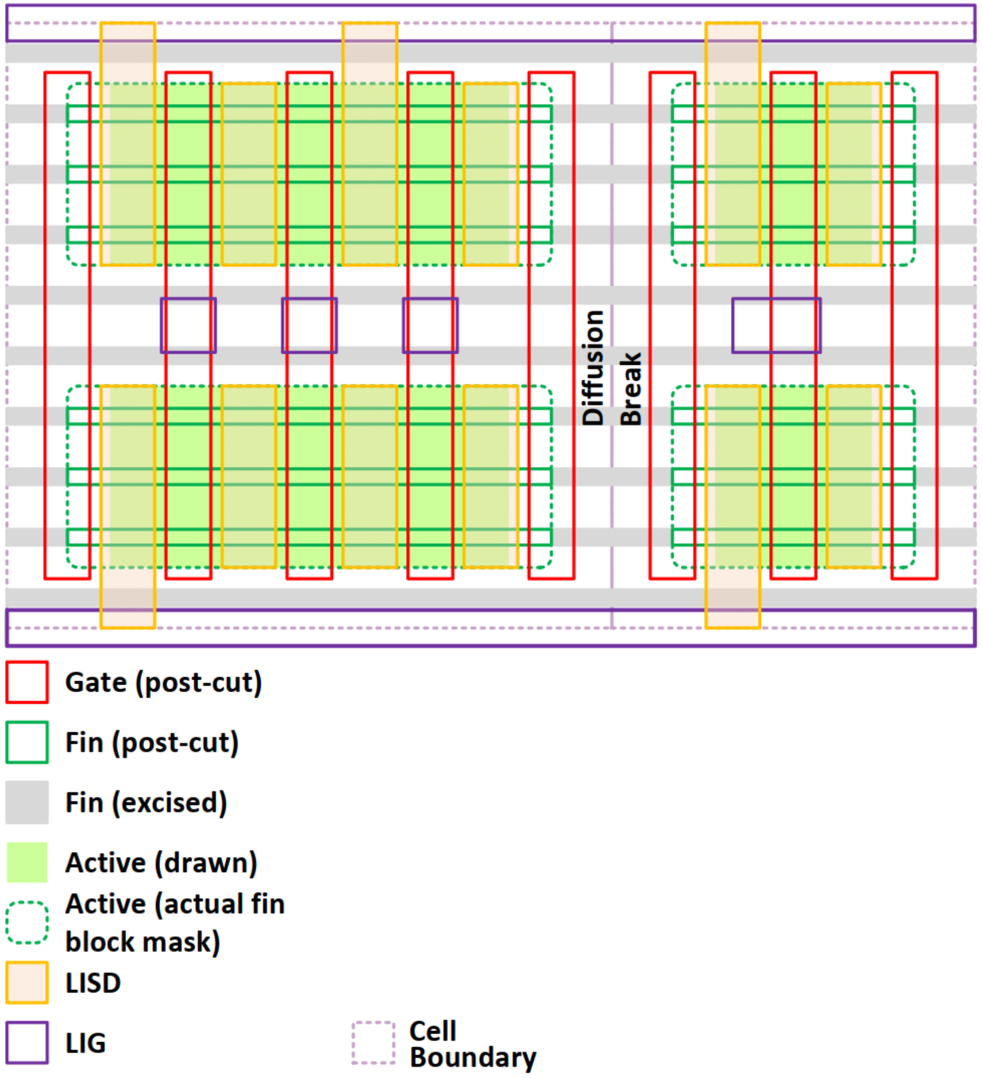
- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**

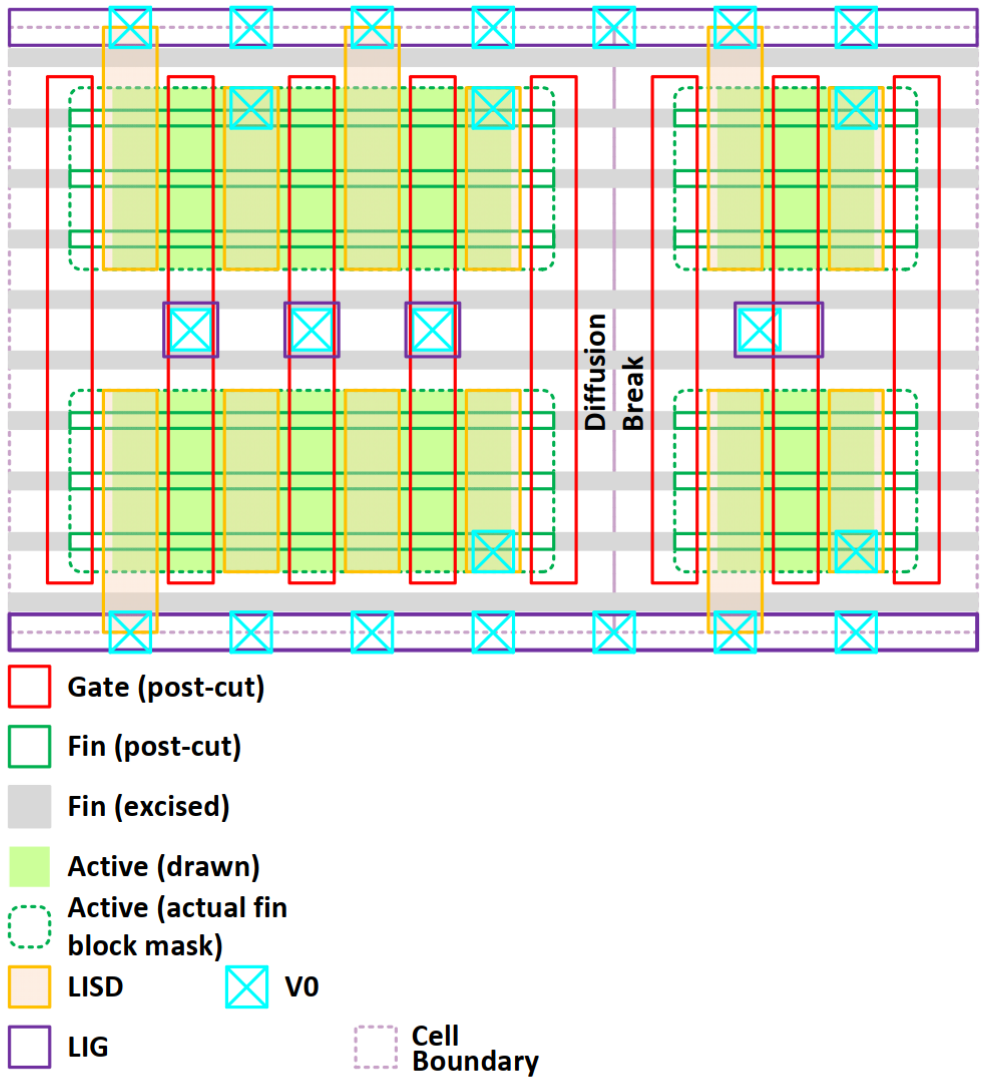
- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

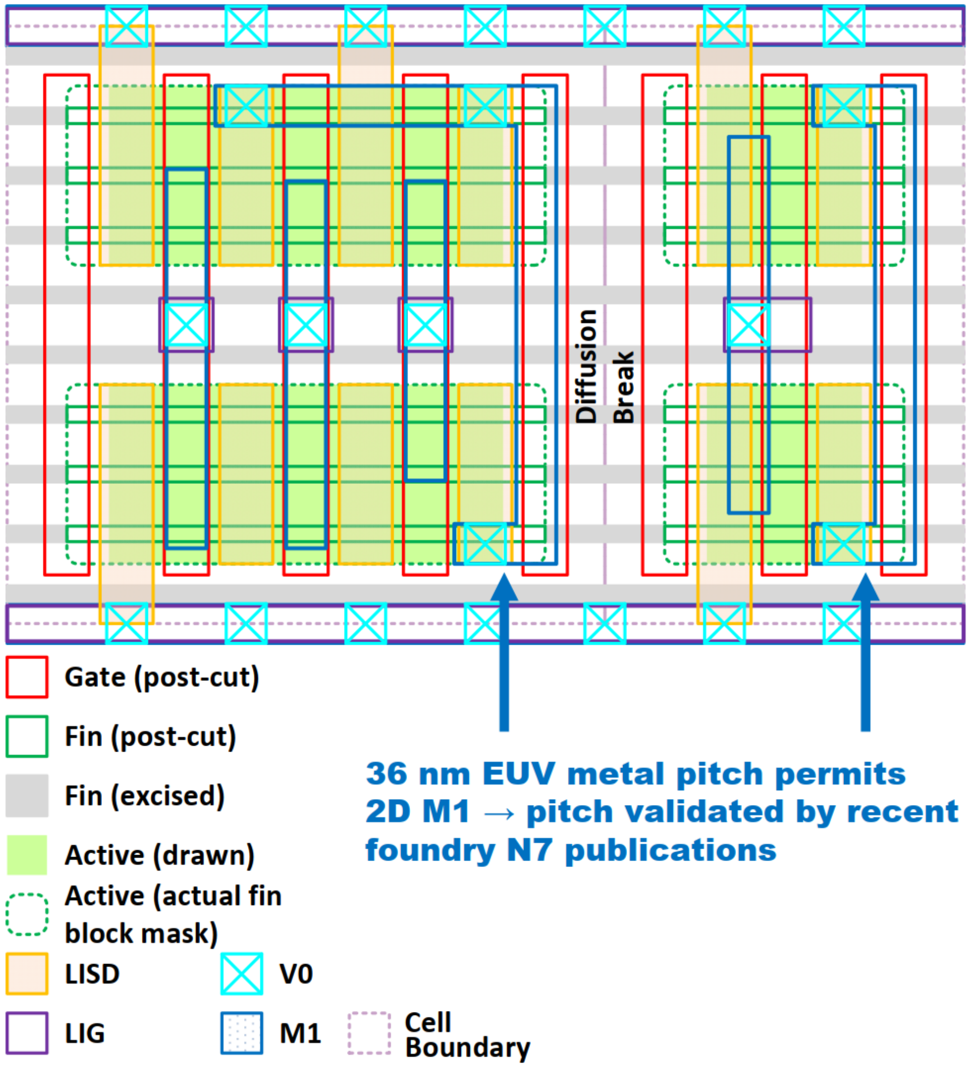
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**

- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

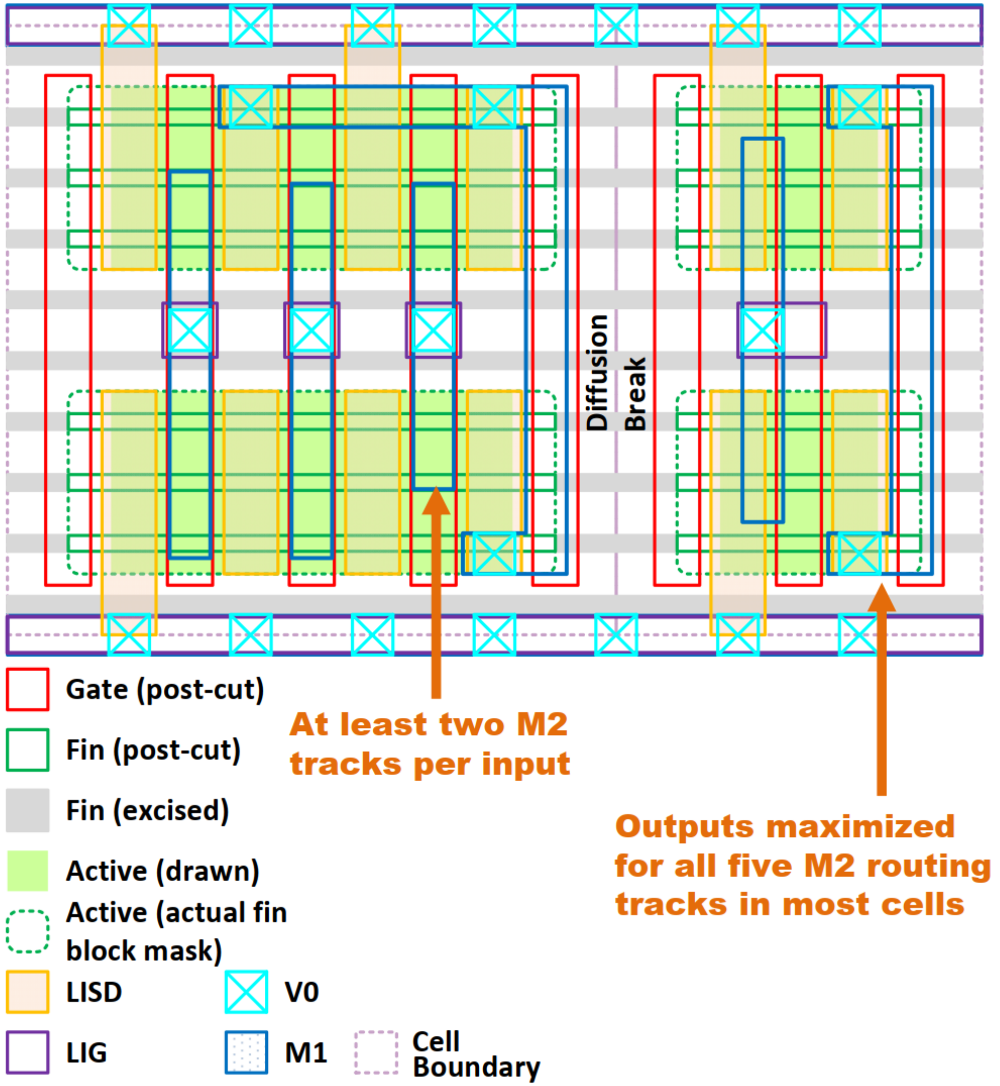
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**
- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**



ASAP7 Standard Cells

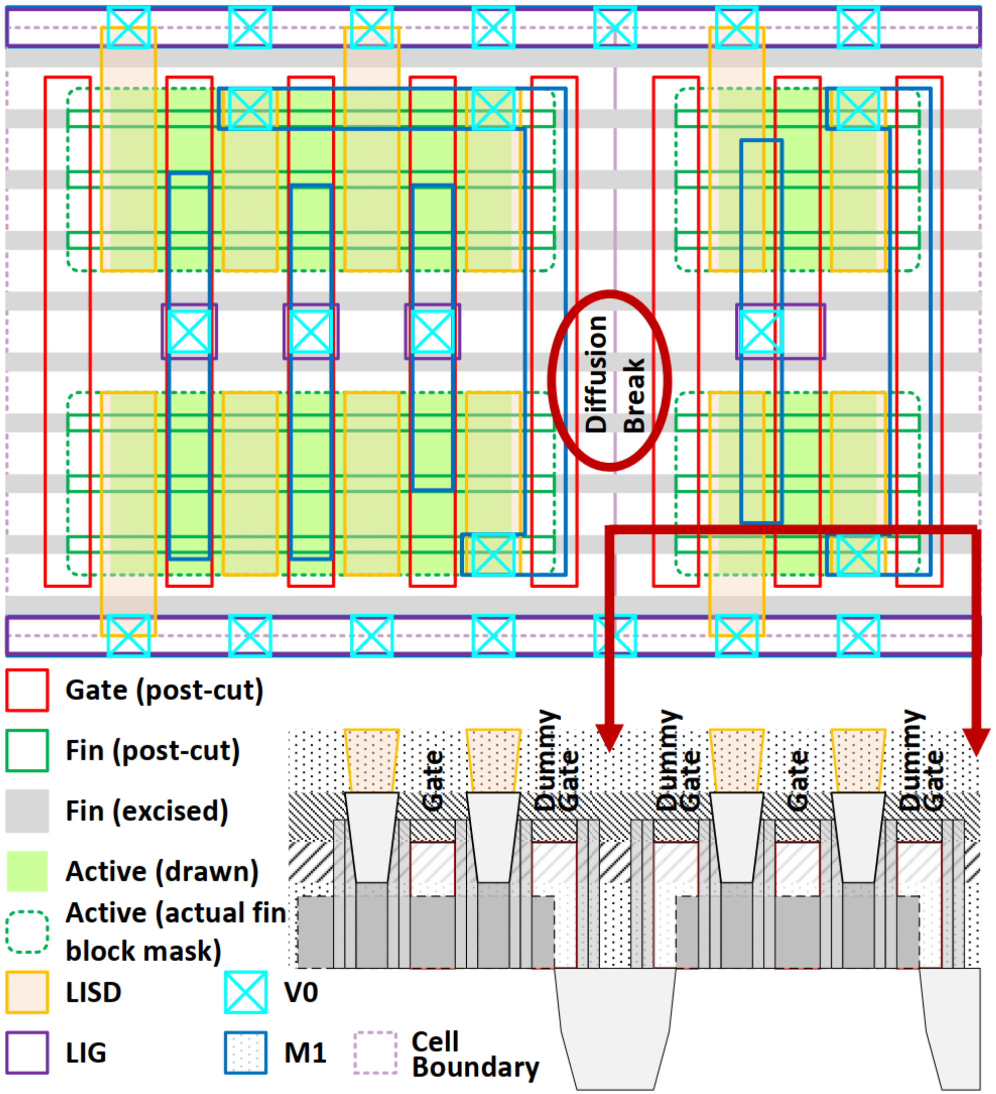
- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**

- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**

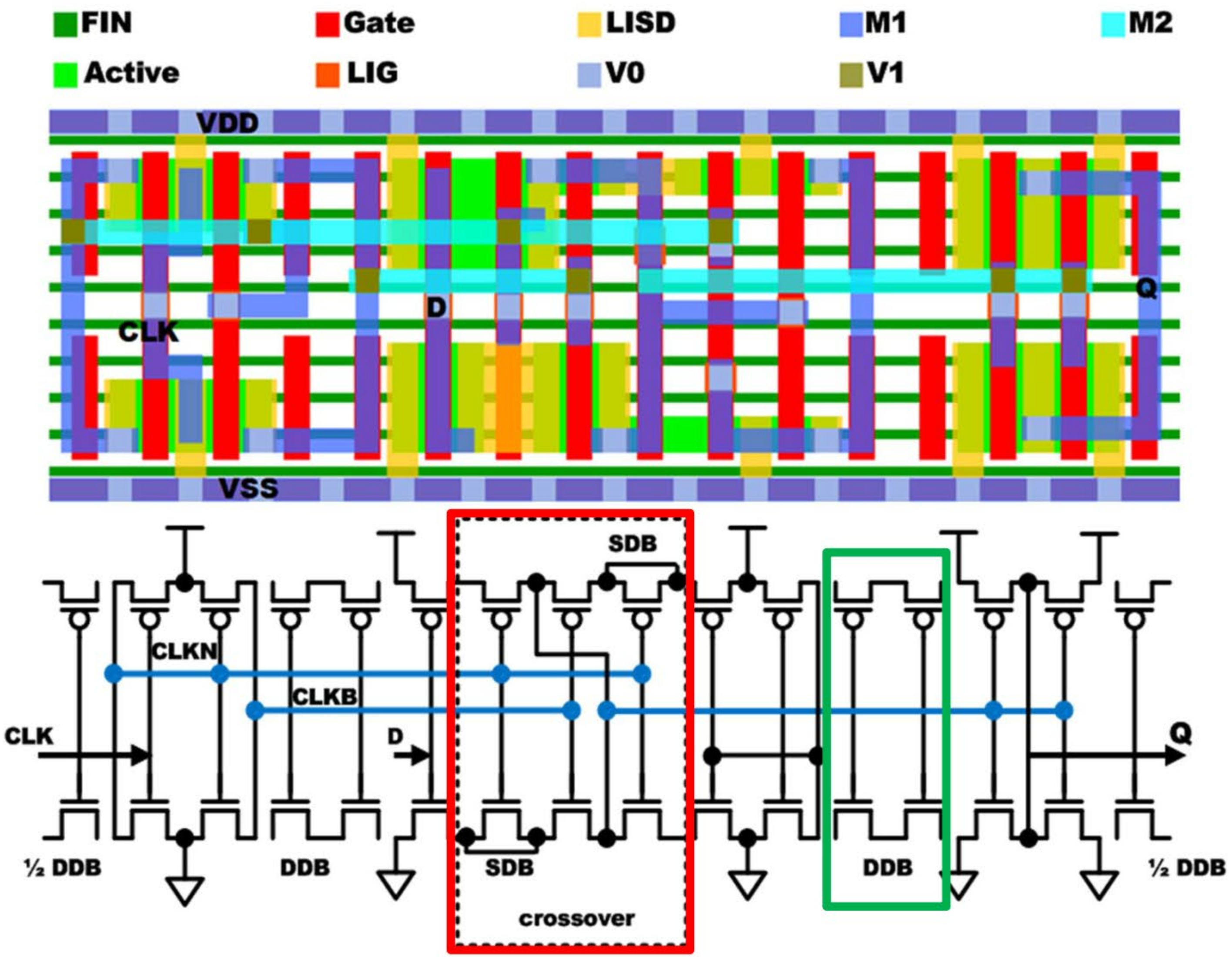


ASAP7 Standard Cells

- **Cell architecture**
  - **7.5 M2 track height**
    - **Provides good gear ratio with fin, poly, and M2 pitch**
  - **Adjacent NAND3 and inverter FEOL and MOL show the double diffusion break (DDB)**
  - **Drawing is not WSYWIG—the fins extend to ½ the gate horizontally past drawn active**
- **DDB needed since the 32 nm node, depending on foundry**
  - **Design rules check for connectivity**

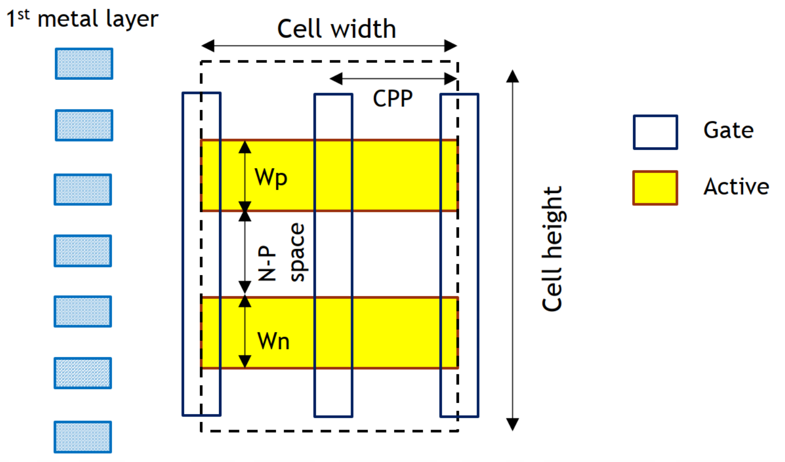


ASAP7 Latch

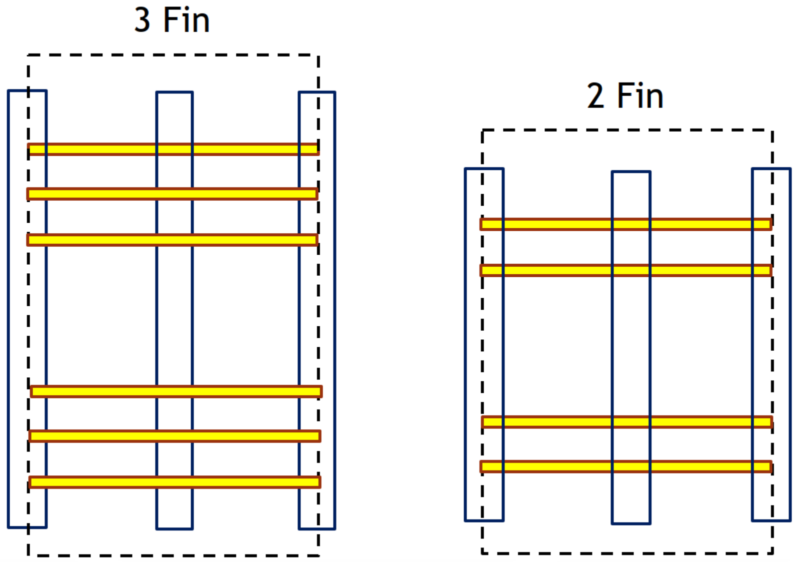


- This demonstrates a crossover
  - Note single diffusion breaks (SBDs)
  - Horizontal M2 can only support limited tracks
- Intel, Samsung support SDBs (no DDBs) at N10/N7 [EETimes]

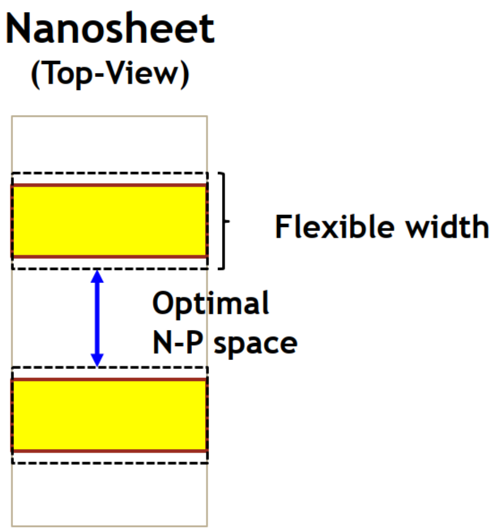
Standard Cell Scaling Beyond N3



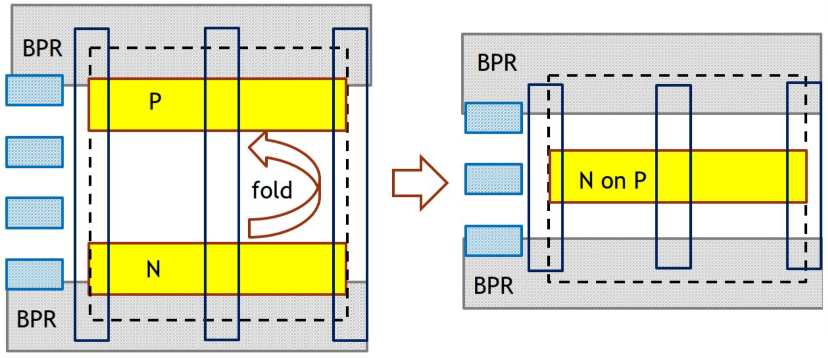
- Fin depopulation



- Nanosheet cell



- Stacked CMOS with buried power rails



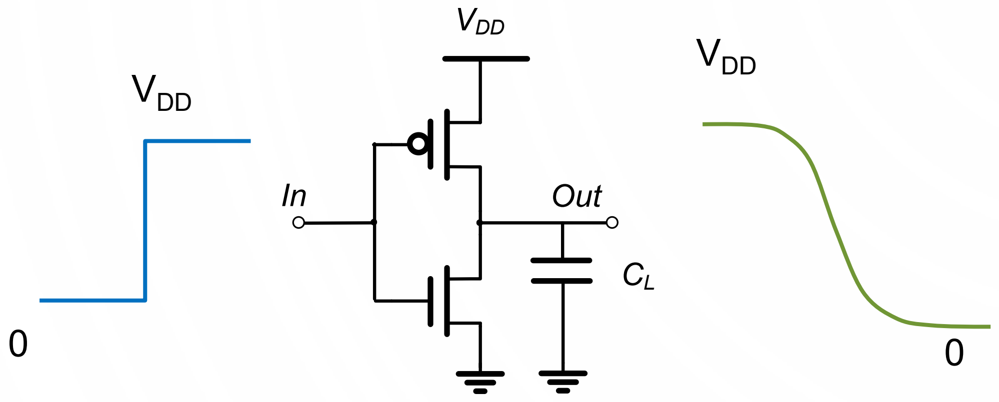
Delay Revisited

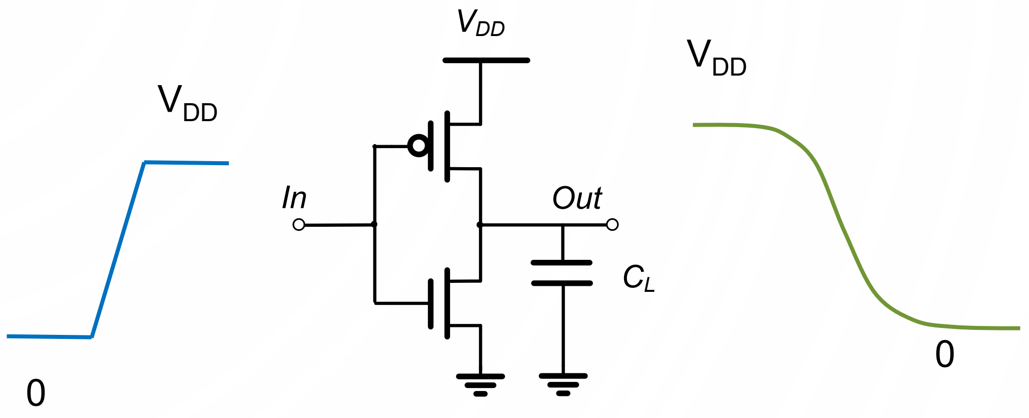
How to Account for Input Slope?

$t_p = \ln 2\ RC = 0.7\ R_{eq}C_L$

$T_{r,10\text{-}90} = 2.2\ R_{eq}C_L$

$T_{r,20\text{-}80} = 1.4\ R_{eq}C_L$



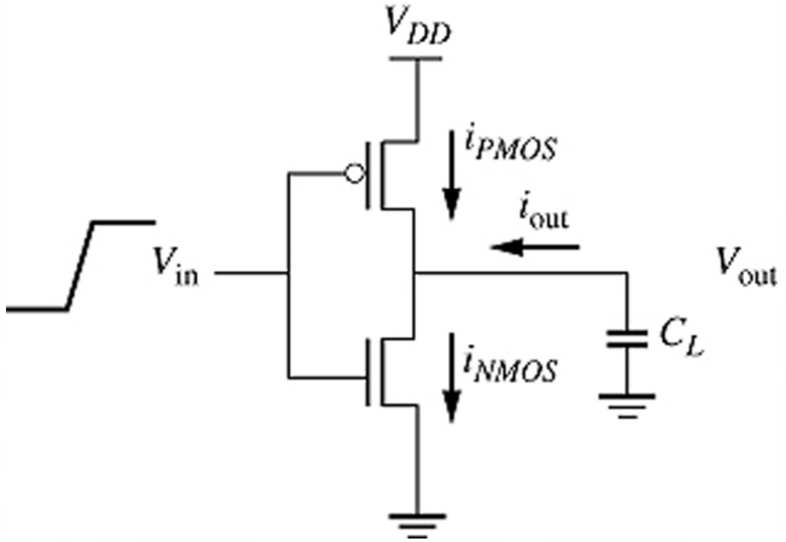


- $t_{pHL} = 0.7\ R_{eq}C_L$

- $t_{pHL} = 0.7\ R_{eq}C_L$

different R_{eq}!

Input Slope Dependence



$$I_{out} = C_L \frac{dV_{out}}{dt} = I_{NMOS} - I_{PMOS}$$

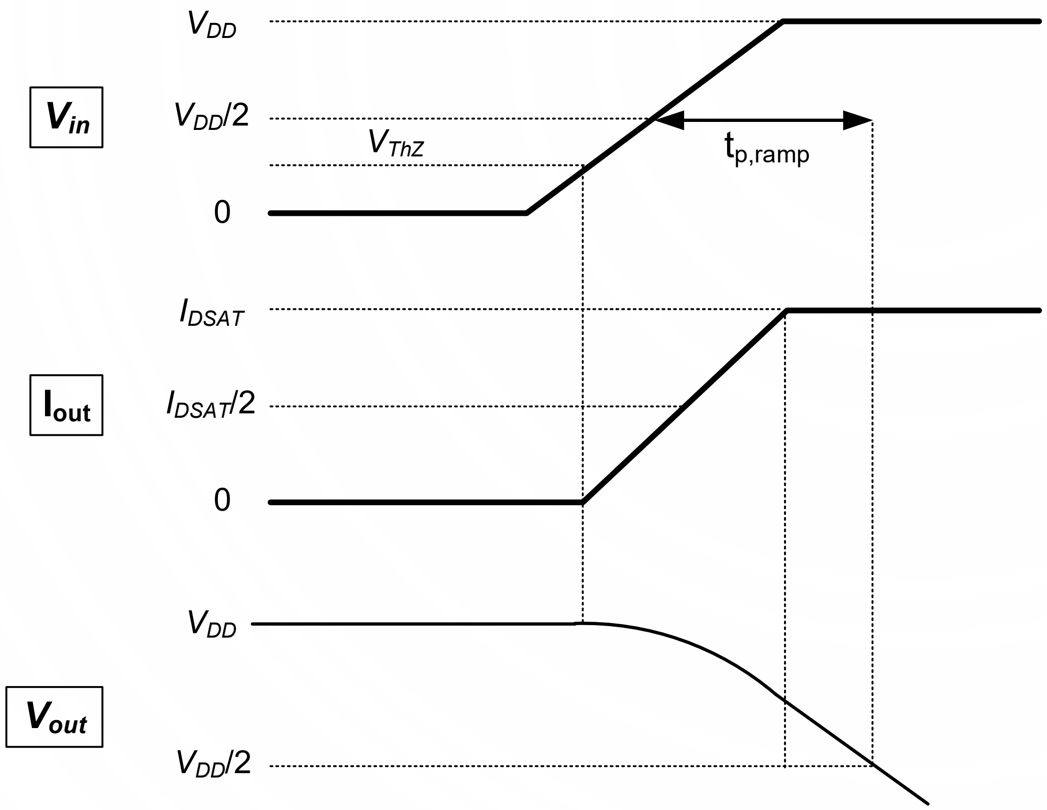
- One way to analyze slope effect
  - Plug non-linear I-V into diff. equation and solve…
- Simpler, approximate solution:
  - Use V_{ThZ} model

Slope Analysis

- For falling edge at output:
  - For reasonable inputs, can ignore I_{PMOS}
  - Either V_{DS} is very small, or V_{GS} is very small

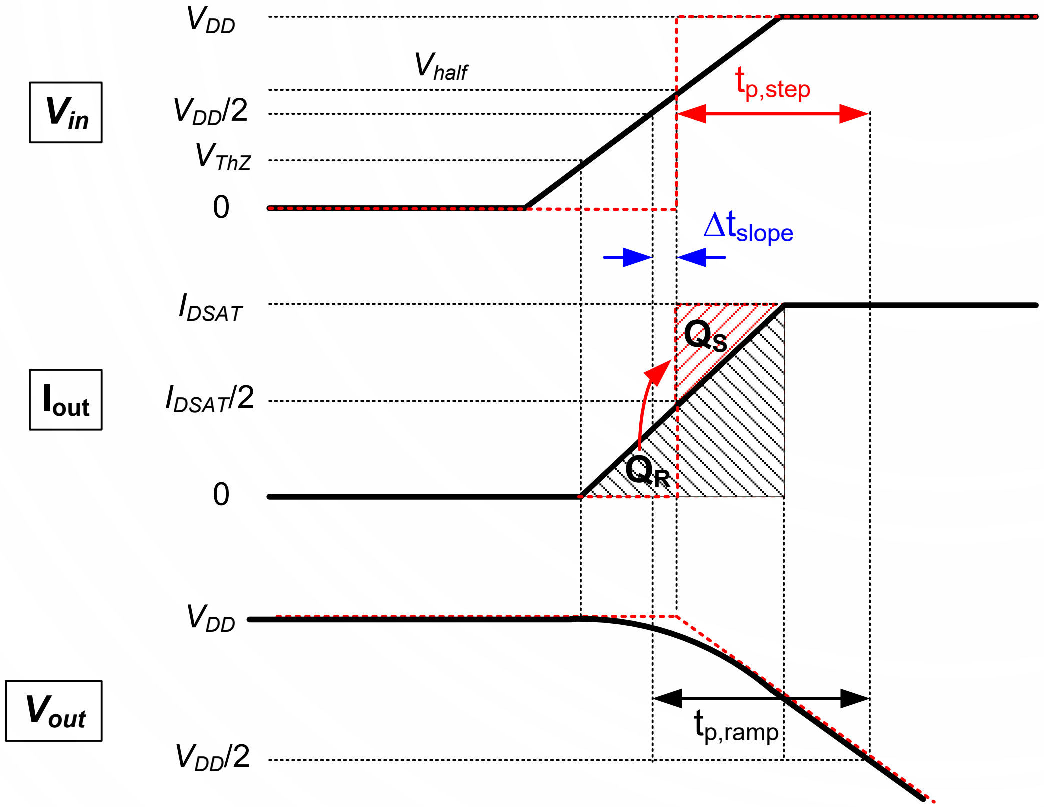
- So, output current ramp starts when $V_{in}=V_{ThZ}$
  - Could evaluate the integral
  - Learn more by using an intuitive, graphical approach

Slope Dependence



- I_{out} ramps linearly for $V_{ThZ}<V_{in}<V_{DD}$
  - Constant once $V_{in}=V_{DD}$
- C_L integrates I_{out}
  - $V_{ThZ}<V_{in}<V_{DD}$: V_{out} quadratic
  - $V_{in} = V_{DD}$: V_{out} linear

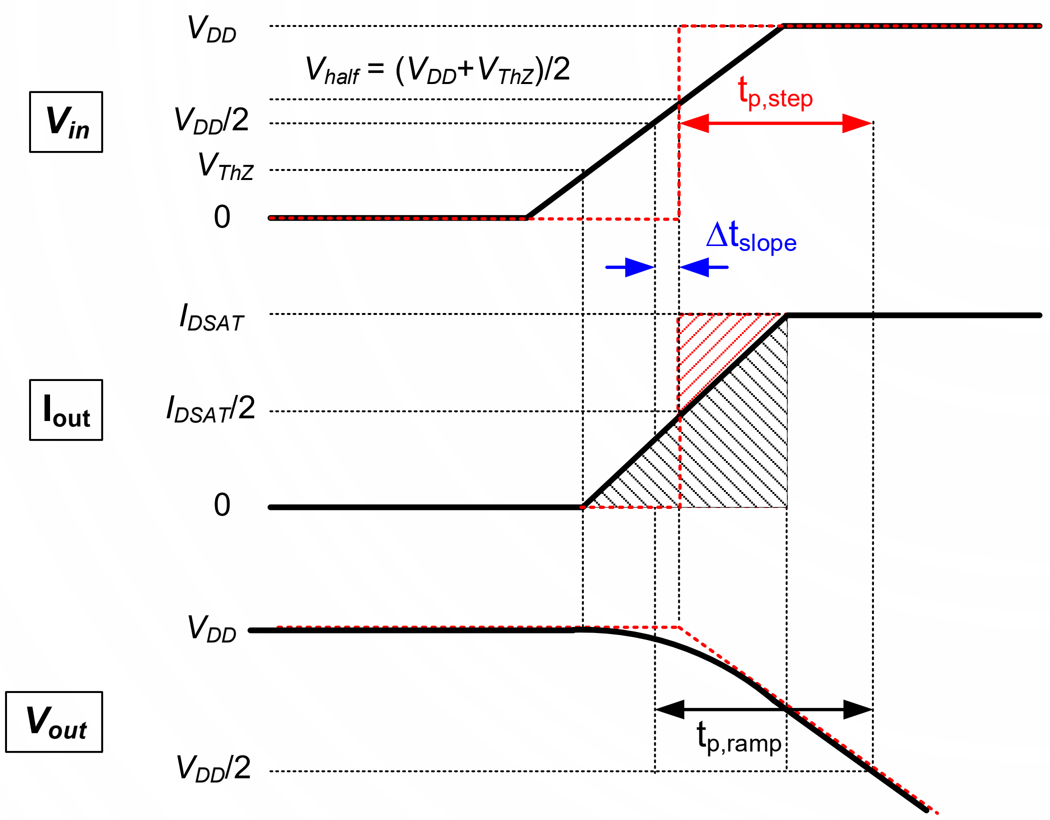
Slope Dependence



From E. Alon

- Compare to step input whose output crosses $V_{DD}/2$ at same time
- V_{out} set by charge removed from C_L
  - Need to make $Q_R = Q_S$
- Step has to shift to when $I_{out}=I_{DSAT}/2$

Slope Dependence



$$t_{p,ramp} = t_{p,step} + \frac{V_{ThZ}/2}{k_r} = t_{p,step} + \frac{V_{ThZ}}{V_{DD}}(1.7t_{p,in})$$

- To find Δt_{slope}:
  - Find $V_{in} = V_{half}$ when $I_{out} = I_{DSAT}/2$
  - And use input t_r
- $I_{DSAT} \sim (V_{DD}-V_{ThZ})$:
  $V_{half} = (V_{DD}+V_{ThZ})/2$
- So $\Delta t_{slope} = (V_{ThZ}/2)/k_r$
  - $k_r = (0.8\text{-}0.2)V_{DD}/(t_{r,20\text{-}80}) = 0.6V_{DD}/(2t_{p,in}) = V_{DD}/(3.3t_{p,in})$
  - $t_{p,in}$ – input propagation delay

Result Summary

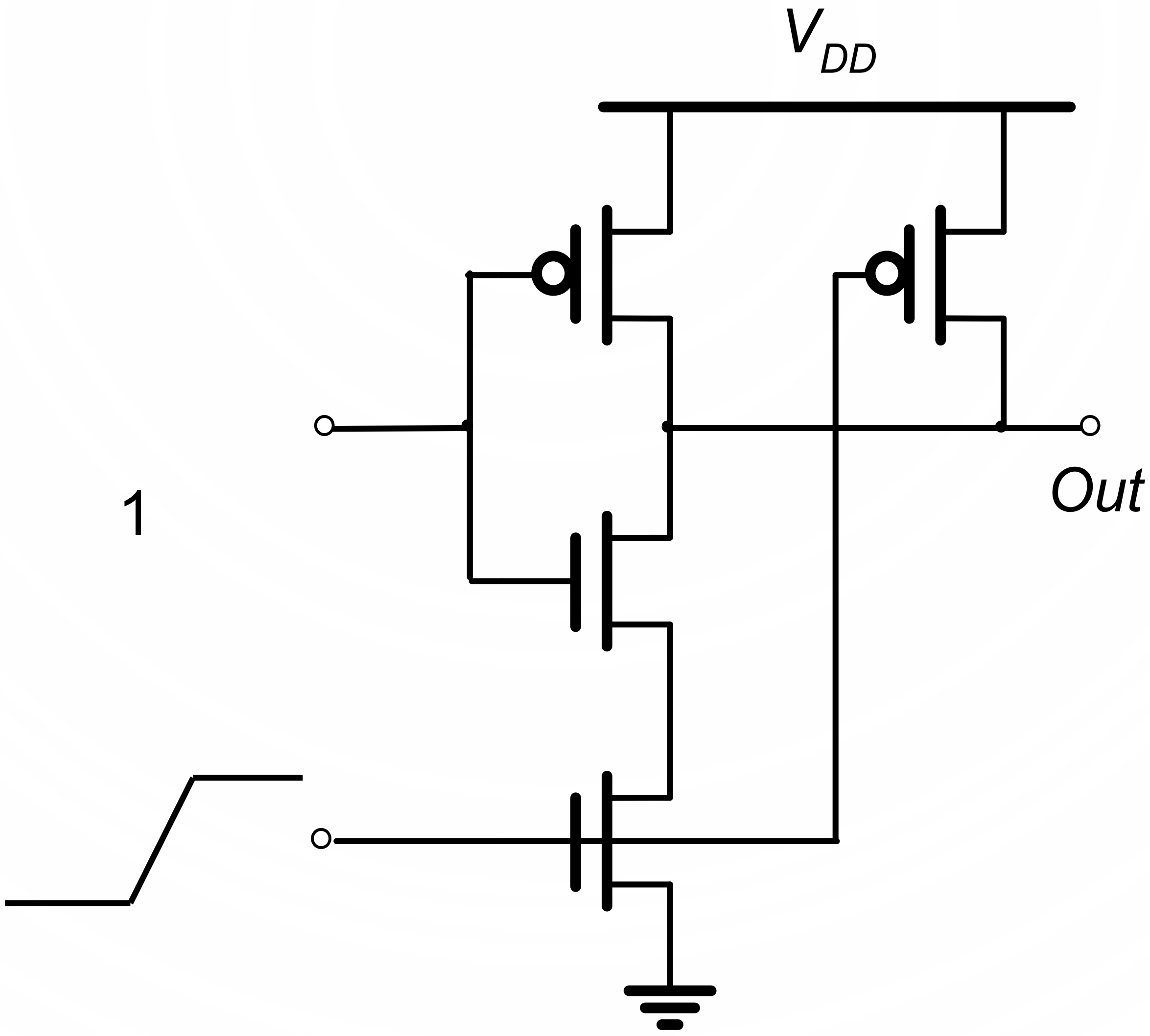
- For reasonable input slopes:

$$t_{p,ramp} = t_{p,step} + \frac{V_{ThZ}/2}{k_r} = t_{p,step} + \frac{V_{ThZ}}{V_{DD}}(1.7t_{p,in}) = t_{p,step} + \frac{V_{ThZ}}{V_{DD}}(0.8t_{r,20-80})$$

- For $t_{p,avg}$: V_{ThZ} is $(V_{ThZN} + V_{ThZP})/2$
  - V_{ThZ}/V_{DD} typically ~1/3-1/2 at nominal supplies
- Propagation delay is a function of
  - Drive strength (R_{eq})
  - Load (C_L)
  - Input rise/fall time (which is proportional to the propagation delay of the previous gate)

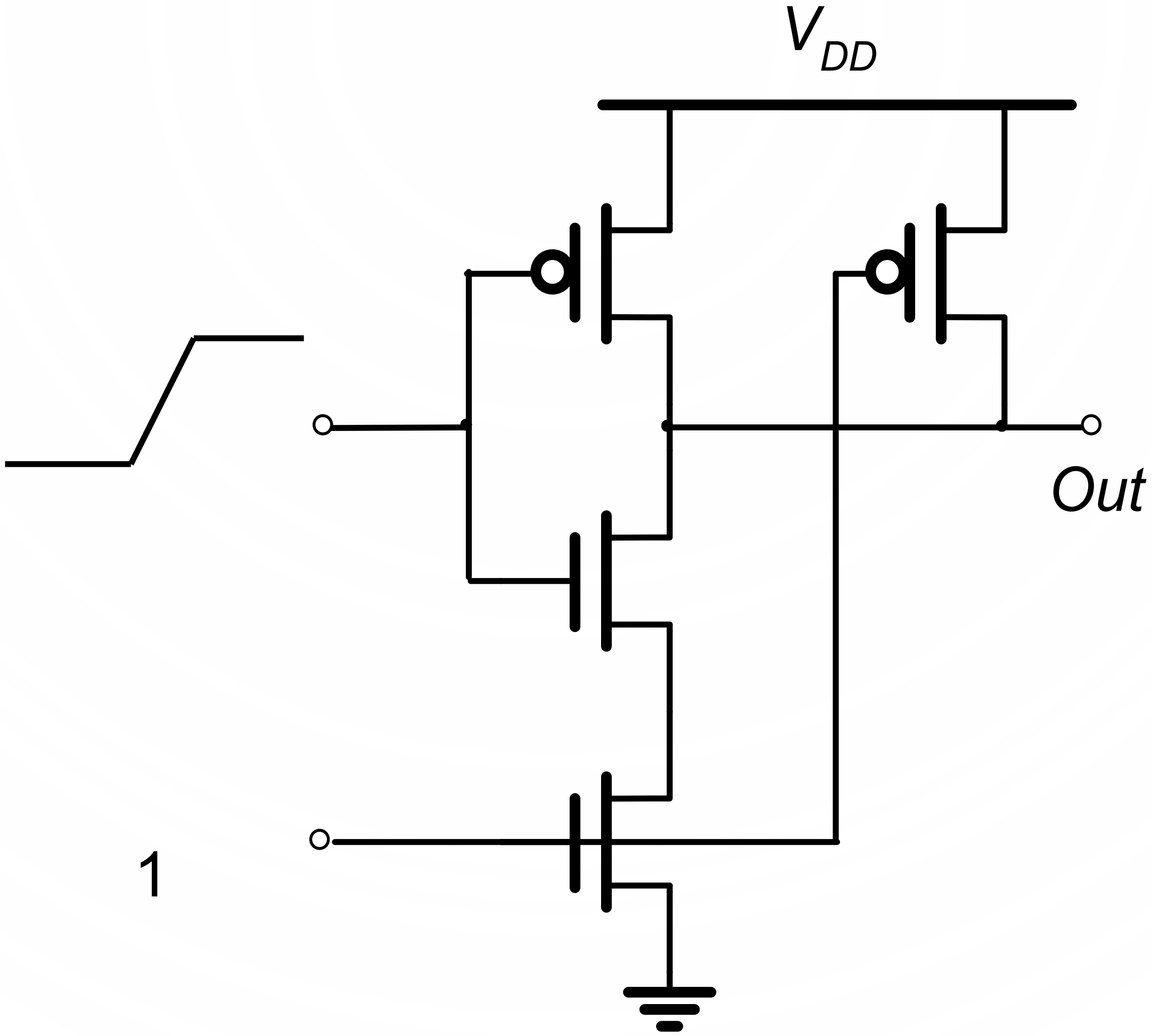
Signal Arrival Times

- NAND gate:



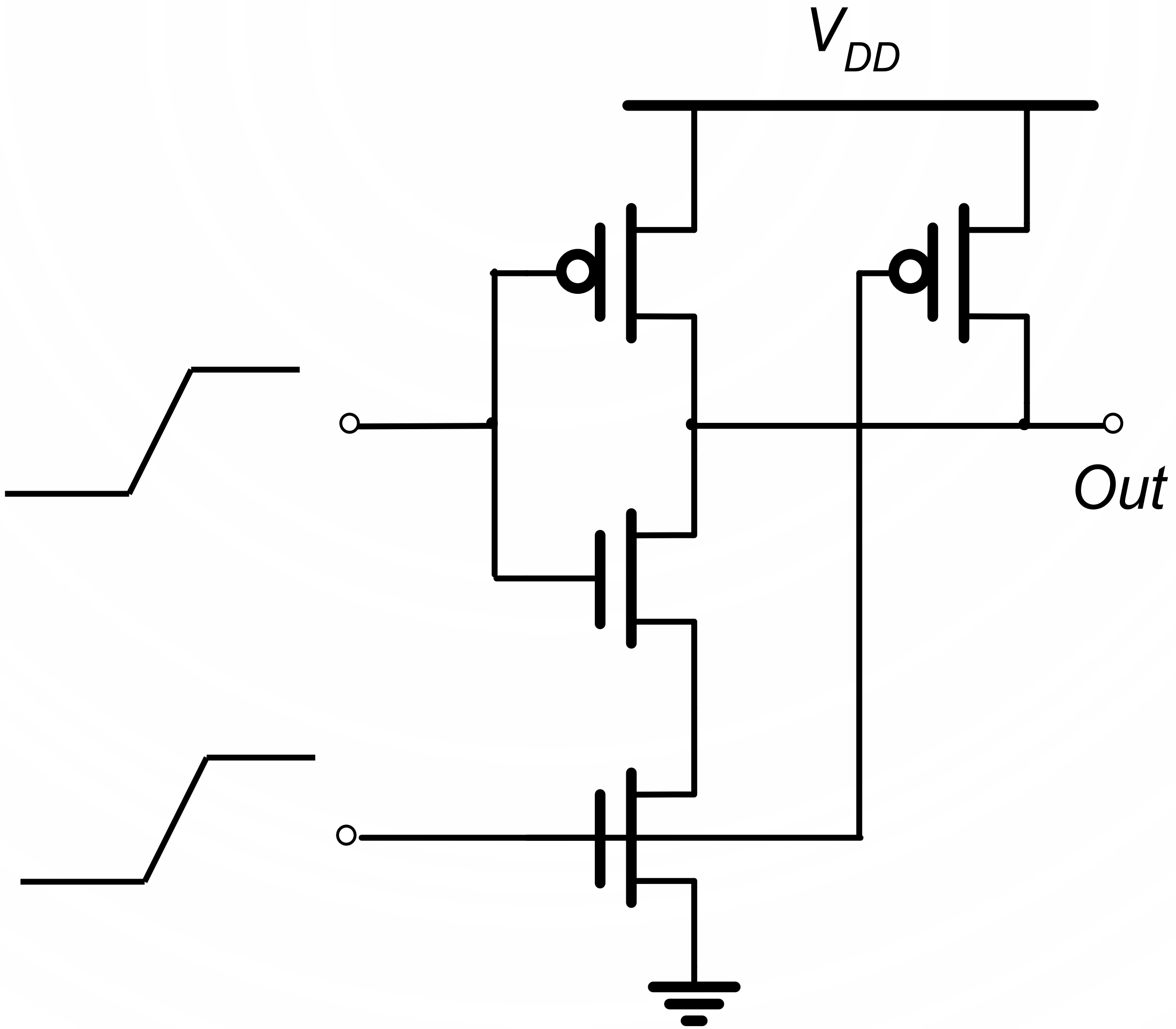
Signal Arrival Times

- NAND gate:

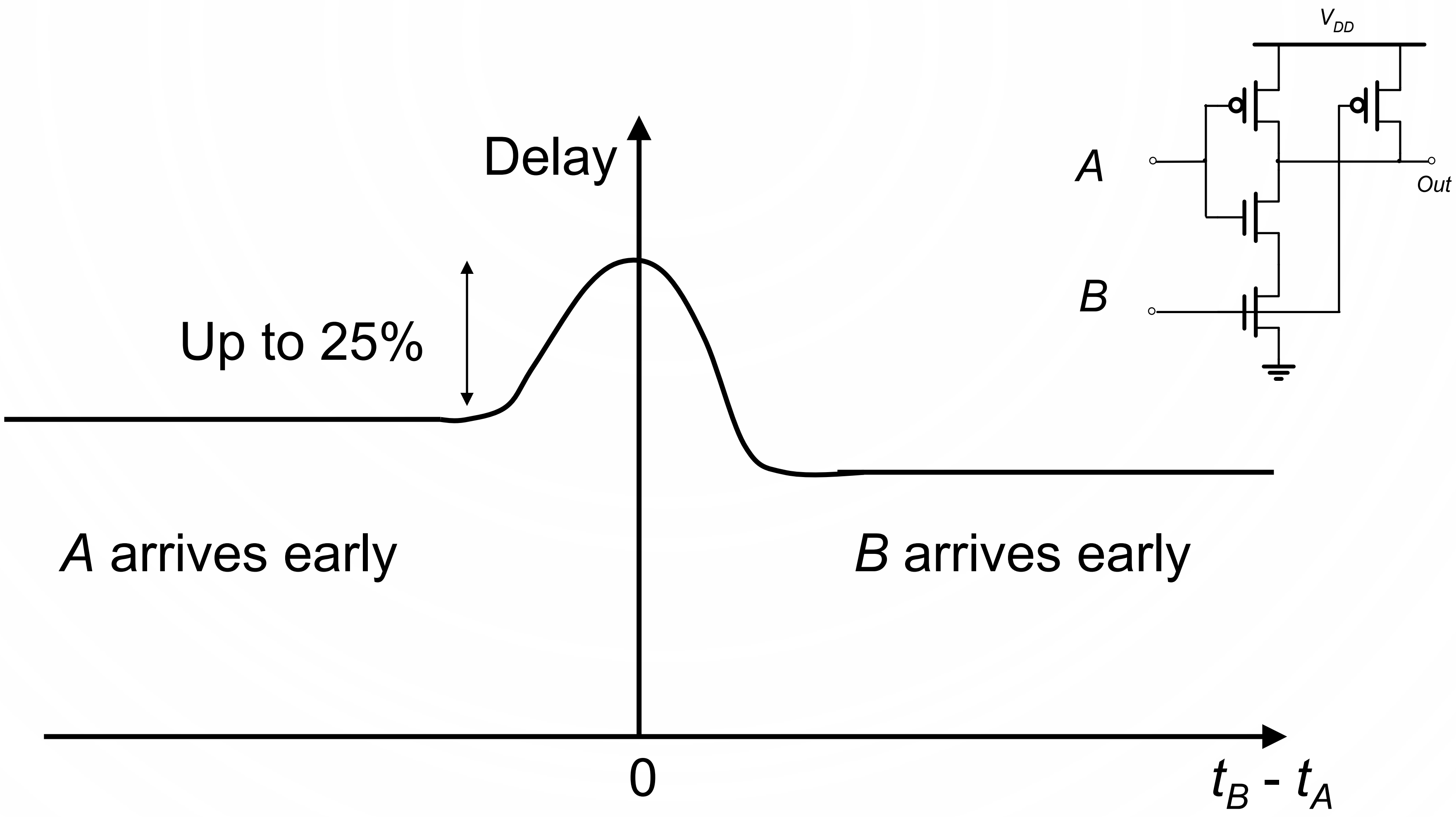


Simultaneous Arrival Times

- NAND gate:



Impact of Arrival Times



The edge can also advance in the opposite transition
Not in models; add derating during design

Standard Cell Library

Standard Cell Library

- Contains for each cell:
  - Functional information: cell = a *b * c
  - Timing information: function of
    - input slew
    - intrinsic delay
    - Input/output capacitance

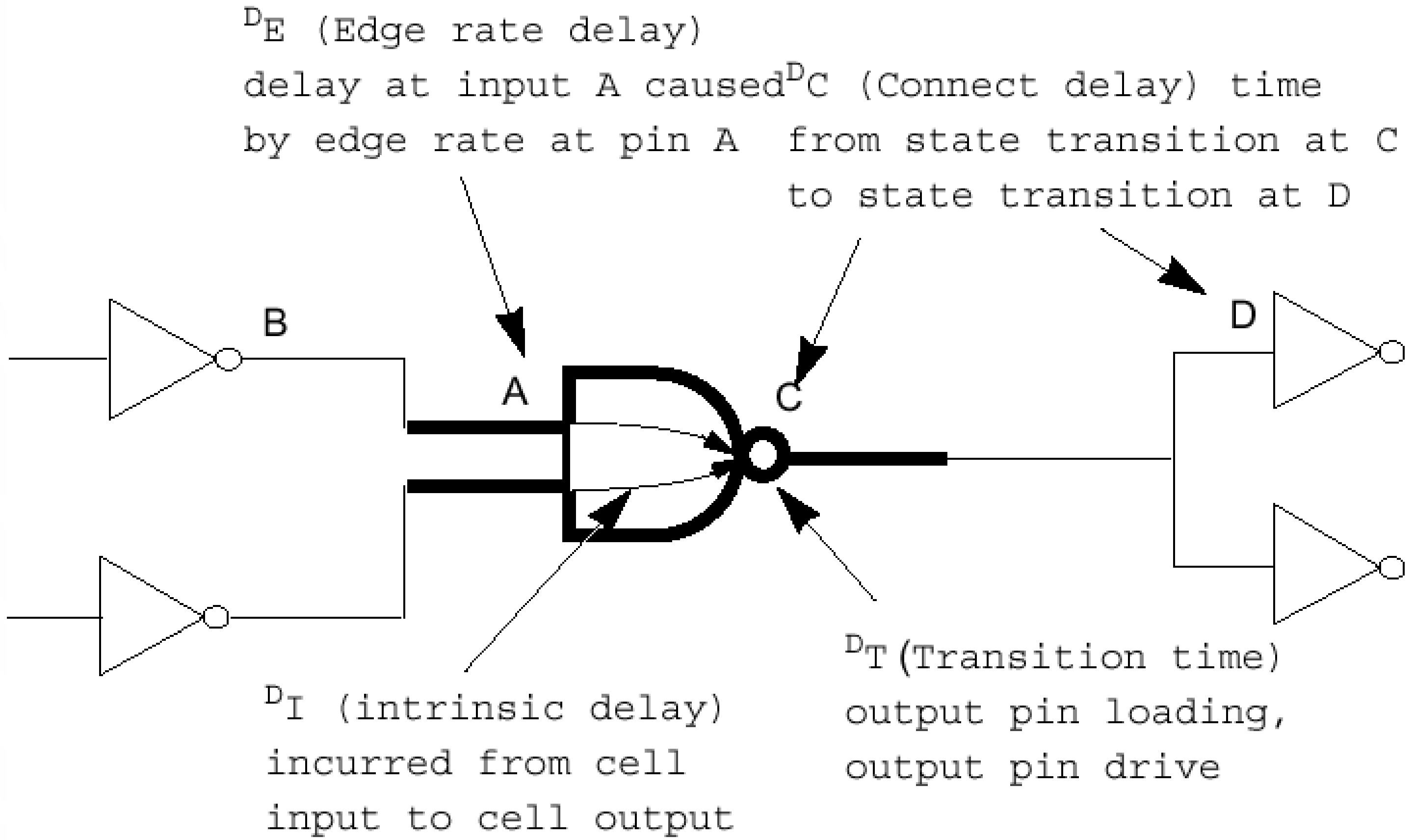
    non-linear models used in tabular approach
  - Physical footprint (area)
  - Power characteristics
  - Noise sensitivity
- Wire-load models - function of
  - Block size
  - Fan-out

Example: NAND2

```
"area": 3.7536,
"cell_footprint": "sky130_fd_sc_hd__nand2",
"cell_leakage_power": 0.00211796,
"driver_waveform_fall": "ramp",
"driver_waveform_rise": "ramp",
"leakage_power": [
  {
    "value": 0.0002796,
    "when": "!A&B"
  },
  {
    "value": 3.005879e-05,
    "when": "!A&!B"
  },
  {
    "value": 0.0079423,
    "when": "A&B"
  },
  {
    "value": 0.0002199,
    "when": "A&!B"
  }
],
"pg_pin,VGND": {
  "pg_type": "primary_ground",
  "related_bias_pin": "VPB",
  "voltage_name": "VGND"
},
"pg_pin,VNB": {
  "pg_type": "nwell",
  "physical_connection": "device_layer",
  "voltage_name": "VNB"
```

Synopsys Delay Models

- Linear (CMOS2) delay model
  - Similar to what we have studied so far



Example Cell Timing

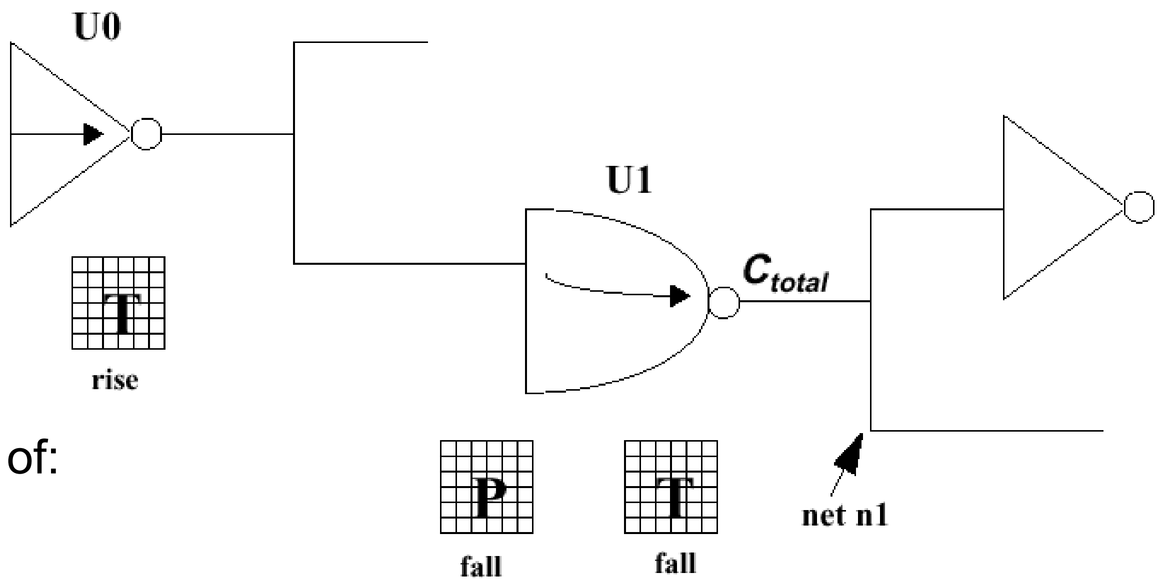
- From Synopsys training materials

```
From pin: U28/A
To pin: U28/Z

arc type :                       cell
arc sense :                      unate
Input net transition times:      Dt_rise = 0.1458, Dt_fall = 0.0653

Rise Delay computation:
rise_intrinsic                   0.48 +
rise_slope * Dt_rise             0 * 0.1458 +
rise_resistance * (pin_cap + wire_cap) / driver_count
0.1443 * (2 + 0) / 1
rise_transition_delay :          0.2886
------------------------------------
Total                            0.7686
```

Cell Characterization (Linear Model)

```
cell(NAND2) {
  area : 1;
  pin(X) {
    function : "(A B)'";
    direction : output;
    edge_rate_rise : 0.24;
    edge_rate_fall : 0.14;
    edge_rate_load_rise : 5.4;
    edge_rate_load_fall : 3.4;
    timing() {
    intrinsic_rise  : 0.34;
    intrinsic_fall  : 0.24;
    rise_resistance : 3.4;
    fall_resistance : 1.4;
    edge_rate_sensitivity_r0 : 0.24;
    edge_rate_sensitivity_f0 : 0.14;
    edge_rate_sensitivity_r1 : 0.14;
    edge_rate_sensitivity_f1 : 0.04;
    related_pin : "A";
    }
    timing() {
    intrinsic_rise  : 0.34;
    intrinsic_fall  : 0.24;
    rise_resistance : 3.4;
    fall_resistance : 1.4;
    edge_rate_sensitivity_r0 : 0.24;
    edge_rate_sensitivity_f0 : 0.14;
    edge_rate_sensitivity_r1 : 0.14;
    edge_rate_sensitivity_f1 : 0.04;
    related_pin : "B";
    }
  }
  pin(A) {
    direction : input;
    capacitance : 0.10;
  }
  pin(B) {
    direction : input;
    capacitance : 0.10;
  }
}
```

(Synopsys) Nonlinear Delay Model (NLDM)



Delay is a function of:

- Rise propagation
- Cell rise
- Fall propagation
- Cell fall
- Rise transition
- Fall transition

NAND2 (Sky130)


```
"timing": [
  {
    "cell_fall,del_1_7_7": {
      "index_1": [
        0.01,
        0.0230506,
        0.0531329,
        0.122474,
        0.282311,
        0.650743,
        1.5
      ],
      "index_2": [
        0.0005,
        0.00131655,
        0.00346659,
        0.00912787,
        0.0240345,
        0.0632852,
        0.166636
      ],
      "values": [
        [
          0.0206305,
          0.0250594,
          0.0363371,
          0.0651531,
          0.1403625,
          0.3379392,
          0.8628026
```

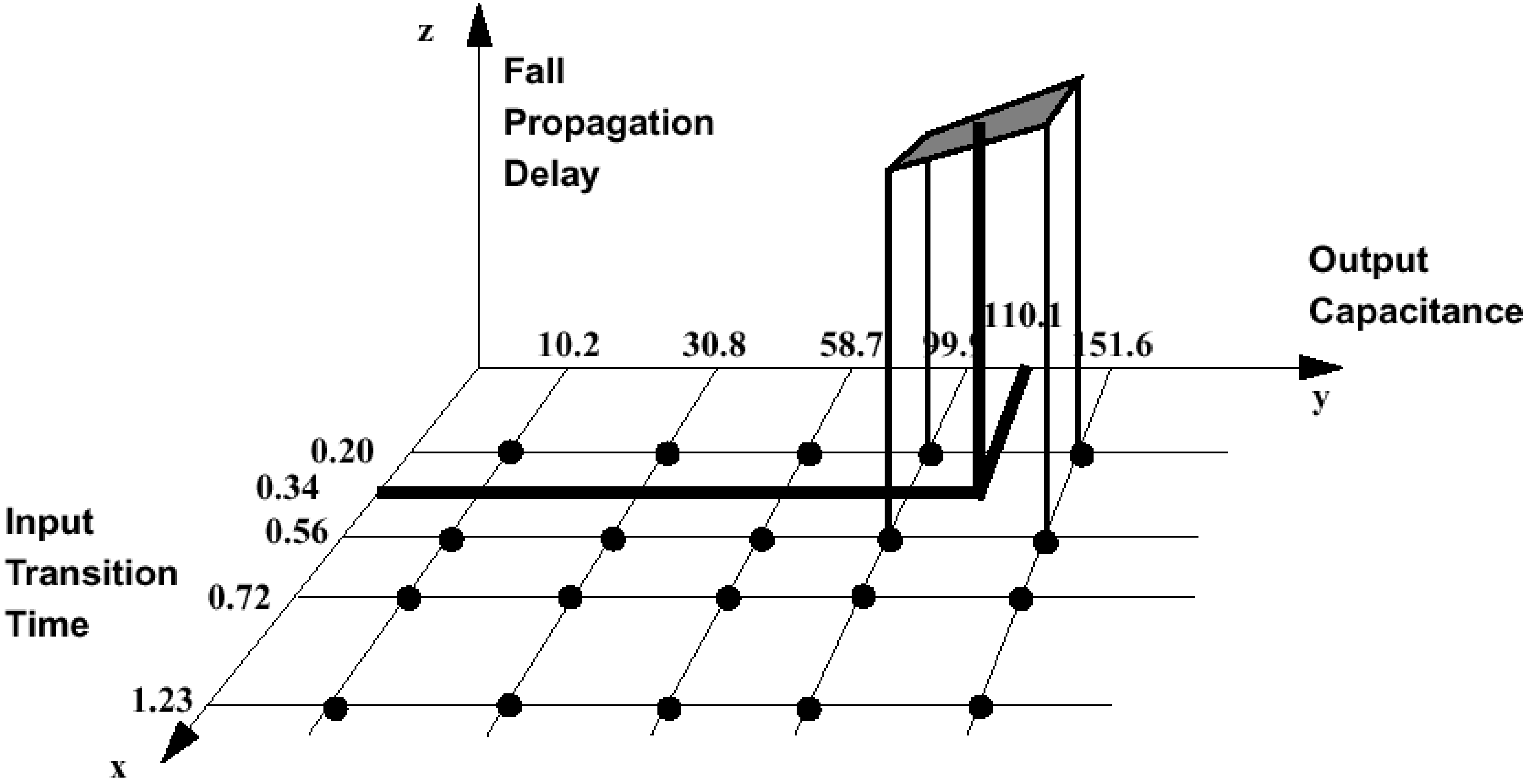
```
"fall_transition,del_1_7_7": {
  "index_1": [
    0.01,
    0.0230506,
    0.0531329,
    0.122474,
    0.282311,
    0.650743,
    1.5
  ],
  "index_2": [
    0.0005,
    0.00131655,
    0.00346659,
    0.00912787,
    0.0240345,
    0.0632852,
    0.166636
  ],
  "values": [
    [
      0.0143751,
      0.0198544,
      0.0342729,
      0.0724393,
      0.1726079,
      0.4374054,
      1.1358902
    ],
    [
      0.0145368,
      0.0198407,
```


Two-dimensional tables of pre-characterized delays/transition times as a function of input slope and output capacitance

Index1 – input transition
Index2 – load capacitance

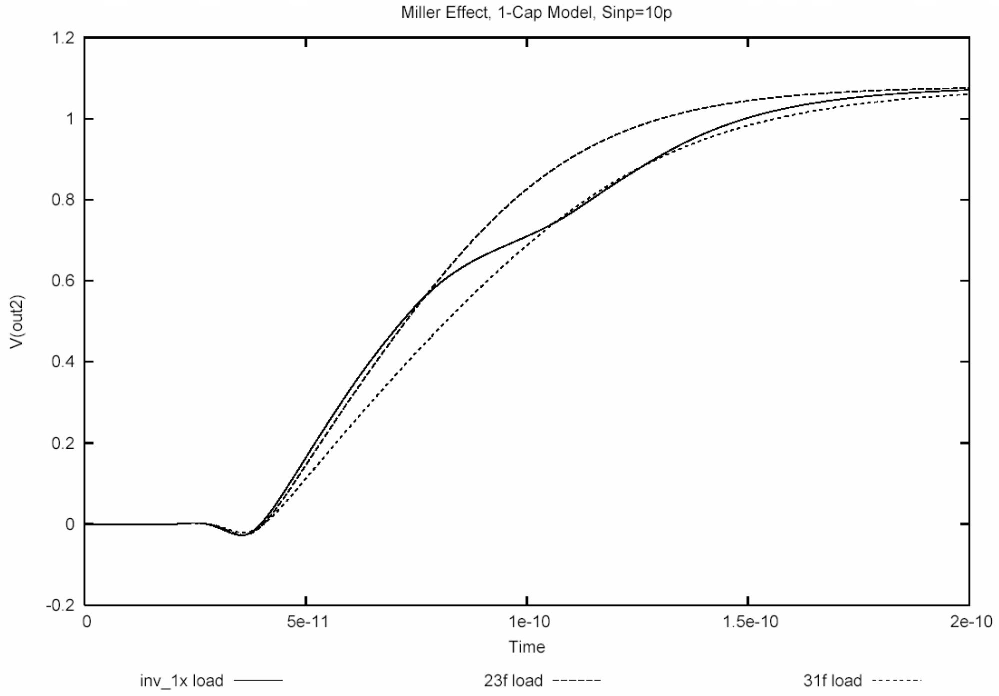
Nonlinear Delay Model (NLDM)

- Interpolates between characterization points

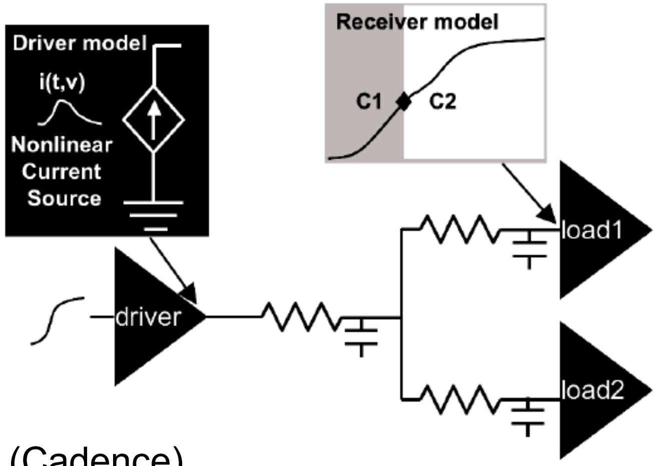


Composite Current Source (CCS) Model

- **Driver model**
  - Composite current source (time and voltage dependent)
- **Receiver model**
  - A set of capacitance models
  - Wire model
- **Interpolate**



Matches both delay and rise/fall times



Synopsys

And then there is Effective Current Source Model … (Cadence)

Summary

- Revisited the delay in CMOS gates
- Analyzed standard cell characterization

Next Lecture

- Revisit timing
  - Latch-based timing