On Friday of last week news broke that someone had infiltrated Nvidia’s network, though at the time it wasn’t clear what they were after or if it was somehow related to Russia’s invasion of Ukraine. It now appears the Nvidia hackers were after a very obvious target: the code behind Nvidia’s Low Hash Rate (LHR) limiter, which reduces GeForce GPU performance when mining cryptocurrency.
Recap

• Linear ID-VGS model is convenient for modeling power-performance around nominal VDD

• Alpha-power law is effective in a wider VDD range

• Transistor leakage is exponential in $V_{Th}$, $V_{DS}$

• When modeling delay, both R and C are linearized
Standard Cells
Standard Cell Inverter

• Schematic and layout
  (in a planar bulk process, \(\sim 250\)nm)

• Pitches are integer multiples of \(\lambda\)
Two Inverters

Share power and ground

Abut cells

Delay is additive
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

V. Vashishta, 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 1/2 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 WSYWYG—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 $\frac{1}{2}$ 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 $\frac{1}{2}$ 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

36 nm EUV metal pitch permits 2D M1 → pitch validated by recent foundry N7 publications
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 WSYWYG—the fins extend to 1/2 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
Design Kit
Design Kit Components

• Physical views
  • Layout and schematic, with abstractions

• Netlist

• Logical view
  • Test view

• Timing, power and noise views

• Documentation
EECS251B Technology

• ASAP7 7nm predictive technology kit
• Multi-vth Standard Cell Library 45 IO pads
• SRAMs
• Design rule manual
Delay Revisited
How to Account for Input Slope?

- \( t_{pHL} = 0.7 \, R_{eq} C_L \)

Different \( R_{eq} \)!
Input Slope Dependence

- One way to analyze slope effect
  - Plug non-linear I-V into diff. equation and solve...

- Simpler, approximate solution:
  - Use $V_{ThZ}$ model

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

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

From E. Alon
Slope Dependence

- To find $\Delta t_{\text{slope}}$:
  - Find $V_{\text{in}} = V_{\text{half}}$ when $I_{\text{out}} = I_{\text{DSAT}}/2$
  - And use input $t_r$

- $I_{\text{DSAT}} \propto (V_{DD} - V_{ThZ})$:
  $$V_{\text{half}} = \frac{(V_{DD} + V_{ThZ})}{2}$$

- So $\Delta t_{\text{slope}} = \frac{(V_{ThZ}/2)}{k_r}$
  - $k_r = \frac{V_{DD}}{(t_{r,20-80})} = \frac{V_{DD}}{(2*t_{p,in})}$
  - $t_{p,in} - \text{input propagation delay}$

\[ t_{p,ramp} = t_{p,\text{step}} + \frac{V_{ThZ}/2}{k_r} = t_{p,\text{step}} + \frac{V_{ThZ}}{V_{DD}}t_{p,in} \]
Result Summary

• For reasonable input slopes:

\[ t_{p,ramp} = t_{p,step} + \frac{V_{ThZ}/2}{k_r} = t_{p,step} + \frac{V_{ThZ}}{V_{DD}} \]

\[ t_{p,avg} = t_{p,step} + \frac{V_{ThZ}}{2V_{DD}} t_{r,20-80} \]

• For \( t_{p,avg} \): \( V_{ThZ} \) is \( (V_{ThZN} + V_{ThZP})/2 \)

  • \( V_{ThZ}/V_{DD} \) typically \( \sim 1/3 - 1/2 \) 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
2. M Standard Cell Library
Standard Cell Library

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

K. Keutzer, EE244
Synopsys Delay Models

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

\[ \Delta E (\text{Edge rate delay}) \]
\[ \Delta T (\text{Transition time}) \]
\[ \Delta D (\text{Connect delay}) \]

\[ \Delta D_I (\text{Intrinsic delay}) \]
\[ \text{delay at input A caused by edge rate at pin A from state transition at C to state transition at D} \]

\[ \text{incurred from cell input to cell output} \]

\[ \text{output pin loading, output pin drive} \]
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";
    }
  }
  pin(A) {
    direction : input;
    capacitance : 0.10;
  }
  pin(B) {
    direction : input;
    capacitance : 0.10;
  }
}
Delay is a function of:

- Rise propagation
- Cell rise
- Fall propagation
- Cell fall
- Rise transition
- Fall transition
Two-dimensional tables of pre-characterized delays/transition times as a function of input slope and output capacitance
Nonlinear Delay Model

- Interpolates between characterization points

![Graph of Nonlinear Delay Model]

- Input Transition Time
- Fall Propagation Delay
- Output Capacitance

EECS251B L12 STANDARD CELLS
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
Summary

- Standard cell architectures depend on technology, maximize density-performance-power

- Standard cell library
  - Delay is tabulated, dependent on slope, load
Next Lecture

• Timing