Elmore Delay

Nachiket Kapre
nachiket@uwaterloo.ca

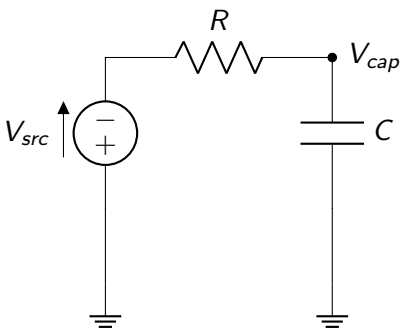
Outline

- Simple RC network
- Elmore Delay Model
- Buffered Interconnect

Delay Modeling

- Approximate delay models useful during FPGA/ASIC CAD optimization
- Chicken-egg problem: How to synthesize circuits for delay before placement/routing?
- Can we predict (to the first order) how fast a given design will run?
  - Can optimize and choose faster circuits during design space exploration
  - Heuristics can use predictions to throw out bad alternatives, focus on better options
- Wire delays are dominant, must model them properly to drive optimizations
- In software, compilers often use **profile-guided optimization** to generate better machine code. Since (1) FPGA CAD tools take very long, and (2) CAD heuristics produce noisy results, this has traditionally not been a sensible strategy.

Analysis of simple RC Networks



- Simple series composition of voltage source + resistor + capacitor
- Apply Ohm's law $V = I \cdot R$, capacitor equation $I = C \cdot \frac{dV}{dt}$, and Kirchoff's voltage law $\sum_i V_i = 0$

$$V_{src} - V_{res} - V_{cap} = 0$$

$$V_{src} - I \cdot R - V_{cap} = 0$$

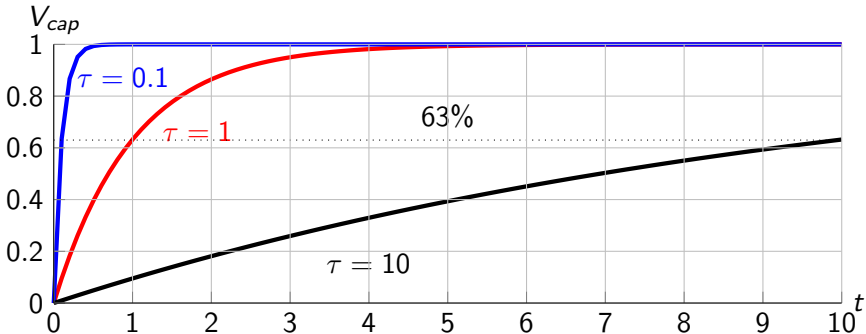
$$V_{src} - C \cdot \frac{dV}{dt} \cdot R - V_{cap} = 0$$

$$V_{cap} = V_{src} - R \cdot C \cdot \frac{dV}{dt}$$

$$V_{cap} = V_{src} \cdot (1 - e^{\frac{-t}{R \cdot C}})$$

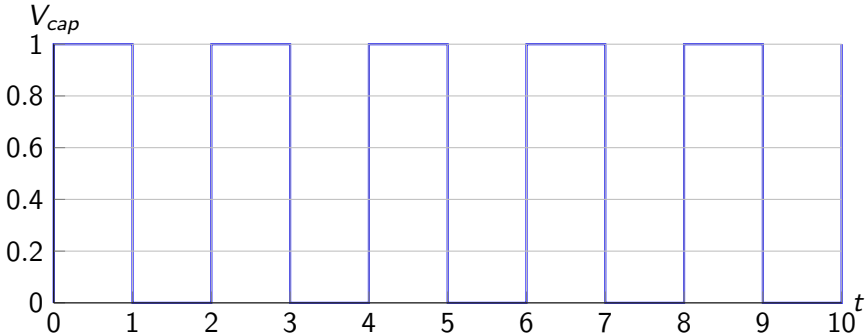
- Time constant $= \tau = R \cdot C$

Impact of τ on charging profiles of capacitor



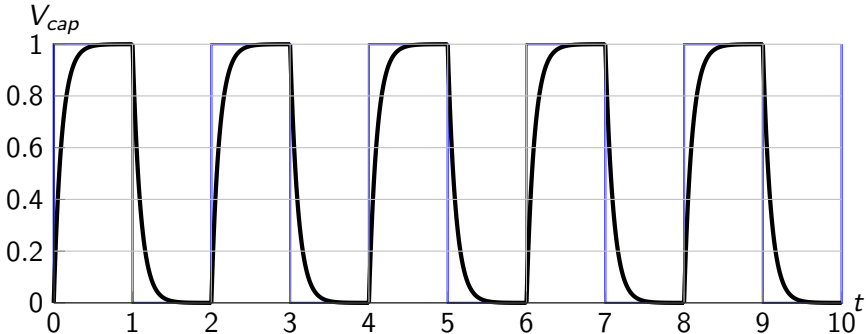
- Different RC time constants deliver different charging profiles
  - Assume unit of $\tau = \Omega \cdot F$, unit of time is s
  - Typically, we design circuits to run at 100s of MHz, so time is in $ns \rightarrow$ RC should be in $k\Omega \cdot pF$ range.
- The RC time constant represents time at which V_{cap} gets 63% of its final value.

Why are we doing this?



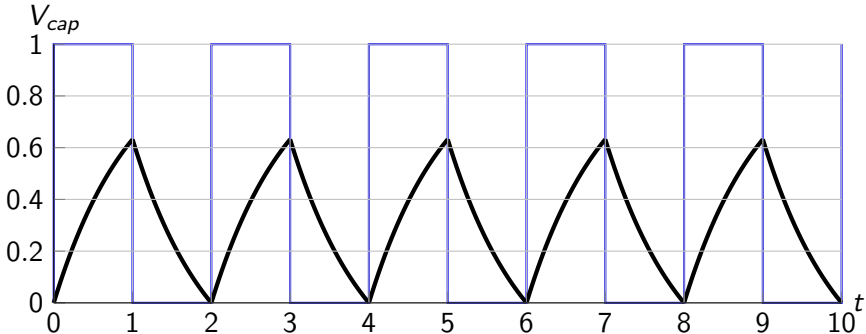
- When drawing timing diagrams we assume square waves, or gently sloping slanting edges
- Real life is *analog* with smoother curves $\rightarrow$ gate voltages have thresholds to detect logic 0s and 1s,
- Can we still deliver proper voltage behavior at gate inputs in the prescribed clock period in presence of analog effects?

Why are we doing this?



- When drawing timing diagrams we assume square waves, or gently sloping slanting edges
- Real life is *analog* with smoother curves $\rightarrow$ gate voltages have thresholds to detect logic 0s and 1s,
- Can we still deliver proper voltage behavior at gate inputs in the prescribed clock period in presence of analog effects?

Why are we doing this?

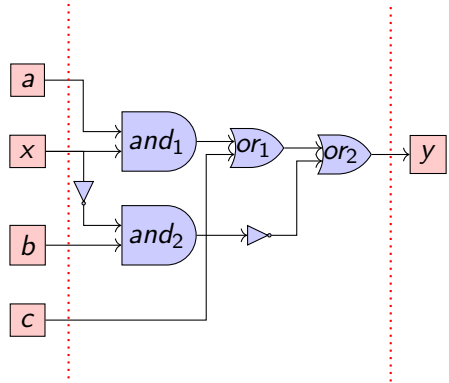


- When drawing timing diagrams we assume square waves, or gently sloping slanting edges
- Real life is *analog* with smoother curves $\rightarrow$ gate voltages have thresholds to detect logic 0s and 1s,
- Can we still deliver proper voltage behavior at gate inputs in the prescribed clock period in presence of analog effects?

Timing Analysis with τ

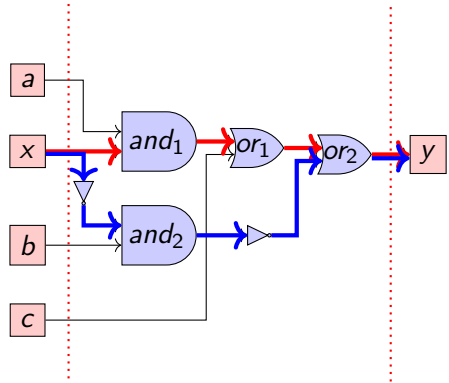
- Static Timing Analysis (STA) identifies longest paths in your circuit and adds up delays of components along the path
  - Exception of false paths which are not activated.
- However, this only considers logic delays
- Modern chips, 80–90% of delay is in wiring
- Wires can be modeled as RC networks $\rightarrow$ but can we estimate delay without solving differential equations?

Static Timing Analysis (revisited)



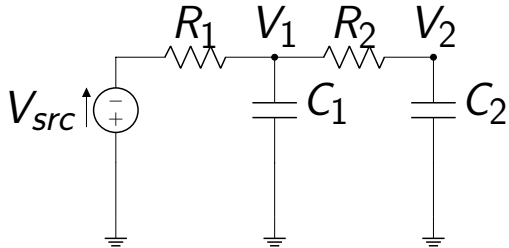
- Using STA, we calculated critical path as the blue edge
- With delay analysis, our job is to annotate each edge with a number corresponding to wire delay
- Possible to consider delay through wire as linearly proportional to length. **Wrong**

Static Timing Analysis (revisited)



- Using STA, we calculated critical path as the blue edge
- With delay analysis, our job is to annotate each edge with a number corresponding to wire delay
- Possible to consider delay through wire as linearly proportional to length. **Wrong**

Analysis of marginally more complex RC Networks



- Apply KVL to find unknown voltage V_2

$$V_{src} - V_{R_1} - V_{R_2} - V_2 = 0 \quad (1)$$

$$V_{src} - (I_1 + I_2) \cdot R_1 - I_2 \cdot R_2 - V_2 = 0 \quad (2)$$

$$V_{src} - I_1 \cdot R_1 - I_2 \cdot (R_1 + R_2) - V_2 = 0 \quad (3)$$

$$V_{src} - C_1 \cdot \frac{dV_1}{dt} \cdot R_1 - C_2 \cdot \frac{dV_2}{dt} \cdot (R_1 + R_2) - V_2 = 0 \quad (4)$$

- No closed form solution exists!

Elmore Delay Analysis

- Interconnect on chips can be modeled as a collection of RC trees $\rightarrow$ need to setup complex differential equations!
- **Solution**: Elmore delay allows us to consider both these effects by assuming that change in voltage at all nodes is identical. *i.e.* $\frac{dV_1}{dt} = \frac{dV_2}{dt}$

$$V_{src} - C_1 \cdot \frac{dV_1}{dt} \cdot R_1 - C_2 \cdot \frac{dV_2}{dt} \cdot (R_1 + R_2) - V_2 = 0$$

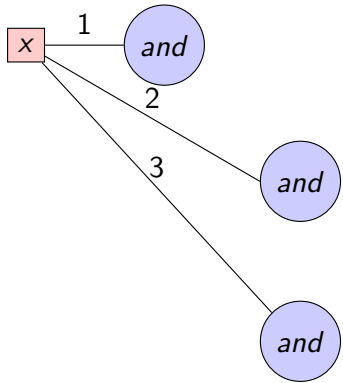
$$V_{src} - C_1 \cdot \frac{dV}{dt} \cdot R_1 - C_2 \cdot \frac{dV}{dt} \cdot (R_1 + R_2) - V_2 \approx 0$$

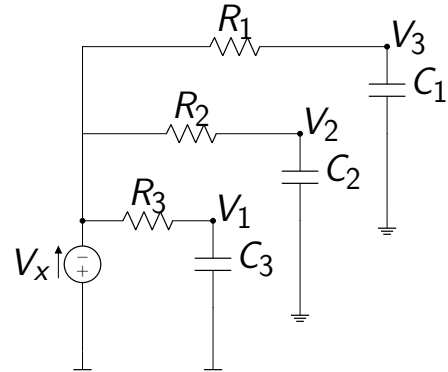
$$V_{src} - \frac{dV}{dt} \cdot (C_1 \cdot R_1 + C_2 \cdot (R_1 + R_2)) - V_2 \approx 0$$

$$V_2 \approx V_{src} - \frac{dV}{dt} \cdot (C_1 \cdot R_1 + C_2 \cdot (R_1 + R_2))$$

$$V_2 \approx V_{src} \cdot (1 - e^{\frac{-t}{C_1 \cdot R_1 + C_2 \cdot (R_1 + R_2)}})$$

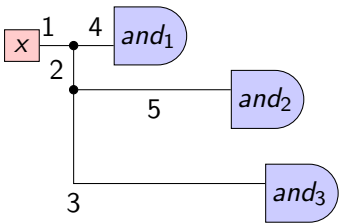
Analyzing a net

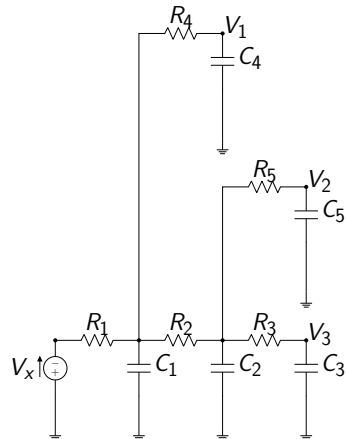




- A **net** is a wire with one source and multiple destinations.
- Replace each wire segment with an RC model
- Each net becomes a **tree** of RC components that **approximately** model the timing behavior of the physical implementation
- Important: Trees do not have cycles

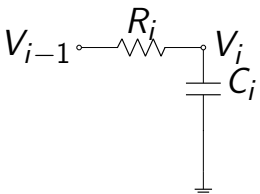
Analyzing a net





- In practice, wire segments are shared during signal routing
- Must decompose the net into a series of segments
- Replace each segment with an RC model

The Elmore Delay Equation

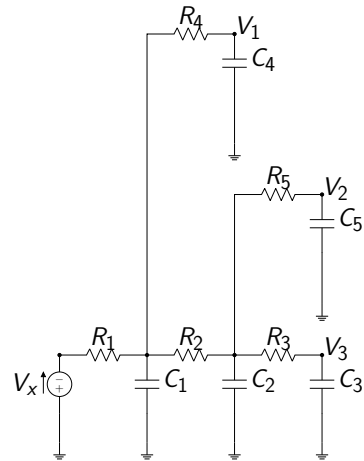


- The RC structure is the building block used in construction of RC trees
  - Model interconnect as a hierarchy of wire segments
  - Common use case: output of gate connected to inputs of several other gates
  - If you find loops, or feedback edges in the resulting network, you are doing something wrong
- To compute delay through R_i, you must consider **all** the capacitance seen at the output node i of the building block
  - For a tree, all sub-tree capacitances must be added. Simply $\sum_k C_k$ for all k connected to i.

- $T_i = T_{R_i} = R_i \cdot \sum_k C_{ik}$
- $T_{path} = \sum_i (R_i \cdot \sum_k C_{ik})$

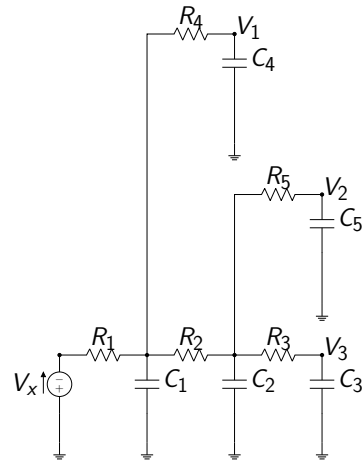
Analyzing a net (substitute values)

- Analyze each segment along $V_x \rightarrow V_3$ path
  - $t_{V_x \rightarrow V_3} = t_{R_1} + t_{R_2} + t_{R_3}$
  - $t_{R_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5)$
  - $t_{R_2} = R_2 \cdot (C_2 + C_3 + C_5)$
  - $t_{R_3} = R_3 \cdot C_3$
- Analyze others paths
  - $t_{V_x \rightarrow V_2} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_2 \cdot (C_2 + C_3 + C_5) + R_5 \cdot C_5$
  - $t_{V_x \rightarrow V_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_4 \cdot C_4$



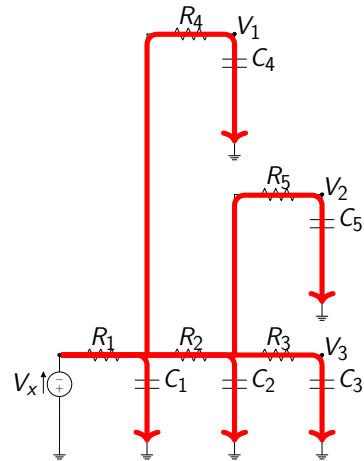
Analyzing a net (substitute values)

- Analyze each segment along $V_x \rightarrow V_3$ path
  - $t_{V_x \rightarrow V_3} = t_{R_1} + t_{R_2} + t_{R_3}$
  - $t_{R_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5)$
  - $t_{R_2} = R_2 \cdot (C_2 + C_3 + C_5)$
  - $t_{R_3} = R_3 \cdot C_3$
- Analyze others paths
  - $t_{V_x \rightarrow V_2} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_2 \cdot (C_2 + C_3 + C_5) + R_5 \cdot C_5$
  - $t_{V_x \rightarrow V_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_4 \cdot C_4$



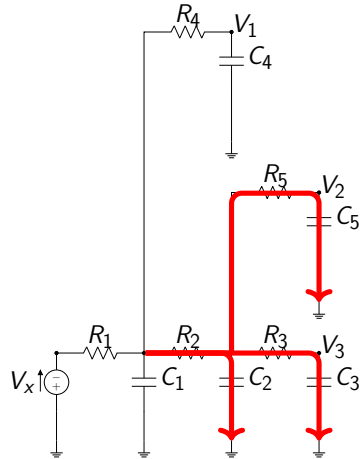
Analyzing a net (substitute values)

- Analyze each segment along $V_x \rightarrow V_3$ path
  - $t_{V_x \rightarrow V_3} = t_{R_1} + t_{R_2} + t_{R_3}$
  - $t_{R_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5)$
  - $t_{R_2} = R_2 \cdot (C_2 + C_3 + C_5)$
  - $t_{R_3} = R_3 \cdot C_3$
- Analyze others paths
  - $t_{V_x \rightarrow V_2} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_2 \cdot (C_2 + C_3 + C_5) + R_5 \cdot C_5$
  - $t_{V_x \rightarrow V_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_4 \cdot C_4$



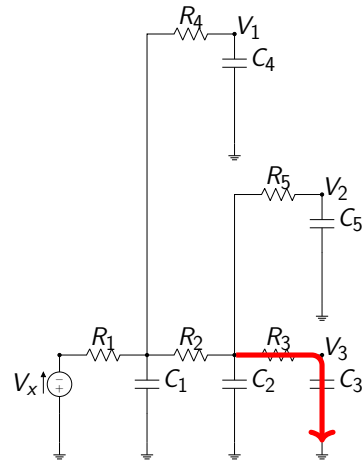
Analyzing a net (substitute values)

- Analyze each segment along $V_x \rightarrow V_3$ path
  - $t_{V_x \rightarrow V_3} = t_{R_1} + t_{R_2} + t_{R_3}$
  - $t_{R_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5)$
  - $t_{R_2} = R_2 \cdot (C_2 + C_3 + C_5)$
  - $t_{R_3} = R_3 \cdot C_3$
- Analyze others paths
  - $t_{V_x \rightarrow V_2} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_2 \cdot (C_2 + C_3 + C_5) + R_5 \cdot C_5$
  - $t_{V_x \rightarrow V_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_4 \cdot C_4$

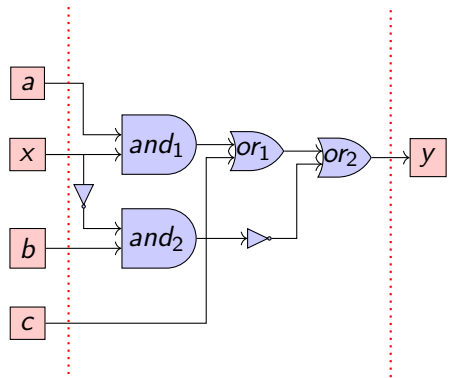


Analyzing a net (substitute values)

- Analyze each segment along $V_x \rightarrow V_3$ path
  - $t_{V_x \rightarrow V_3} = t_{R_1} + t_{R_2} + t_{R_3}$
  - $t_{R_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5)$
  - $t_{R_2} = R_2 \cdot (C_2 + C_3 + C_5)$
  - $t_{R_3} = R_3 \cdot C_3$
- Analyze others paths
  - $t_{V_x \rightarrow V_2} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_2 \cdot (C_2 + C_3 + C_5) + R_5 \cdot C_5$
  - $t_{V_x \rightarrow V_1} = R_1 \cdot (C_1 + C_2 + C_3 + C_4 + C_5) + R_4 \cdot C_4$

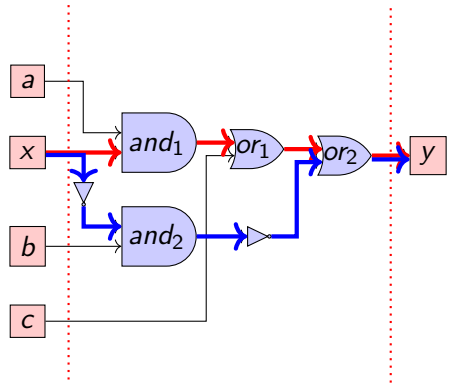


Static Timing Analysis (re-revisited)



- When assuming equa delay per gate, and zero wire delay, we may think the blue critical path goes through the most number of gates → 2 NOT gates, 1 AND gate + 1 OR gate
- After Elmore delay analysis of paths, maybe $x \rightarrow and_1$ connection makes that the critical path!

Static Timing Analysis (re-revisited)

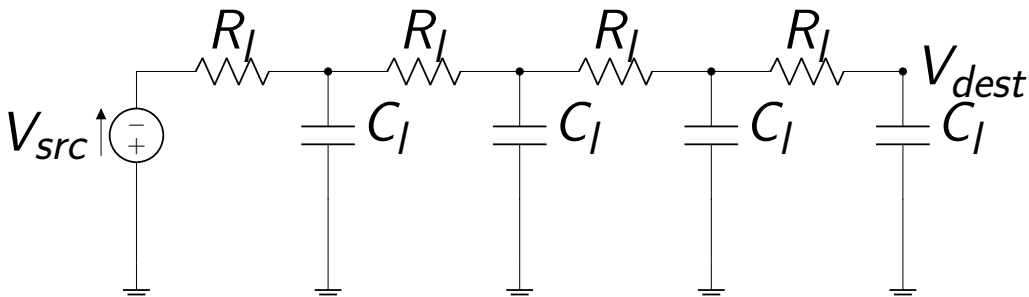


- When assuming equa delay per gate, and zero wire delay, we may think the blue critical path goes through the most number of gates → 2 NOT gates, 1 AND gate + 1 OR gate
- After Elmore delay analysis of paths, maybe $x \rightarrow and_1$ connection makes that the critical path!

Buffered Interconnect

- Wire delays are dominant in modern chips – FPGAs and ASICs
- A chip typically is a 3D stack of multiple metal layers + logic layers at the bottom
- Shorter wires typically use lower levels for routing, Longer wires in upper levels
- Wire cross sections look pretty unusual in modern chips

RC model of long wires



- We know long wires are slower than shorter wires, but by how much?
- Long wires can be broken down into series of RC blocks
- Each RC block models the delay behavior of a *unit length* wire segment
- R_l and C_l are resistance and capacitance per unit length

RC model of long wires

- Applying Elmore delay model to this network with N segments, we can write:
- For first segment, $T_0 = R_l \cdot \sum_{i=0}^{i=N-1} C_l$ generalizes to $\rightarrow T_j = R_l \cdot \sum_{i=j}^{i=N-1} C_l$
- $T_{V_{src} \rightarrow V_{dest}} = R_l \sum_{i=0}^{i=N-1} C_l + R_l \sum_{i=1}^{i=N-1} C_l + R_l \sum_{i=2}^{i=N-1} C_l + \ldots + R_l \sum_{i=N-1}^{i=N-1} C_l$

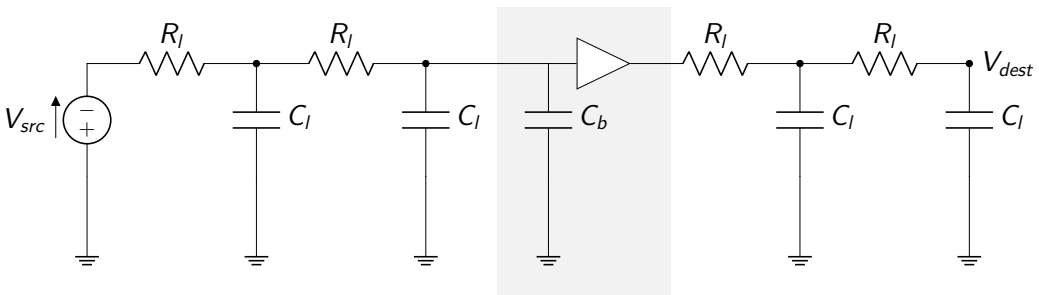
$$
\begin{aligned}
T_{V_{src} \rightarrow V_{dest}} &= \sum_{j=0}^{j=N-1} T_j \\
&= \sum_{j=0}^{j=N-1} R_l \sum_{i=j}^{i=N-1} C_l \\
&= R_l \cdot C_l \sum_{j=0}^{j=N-1} \sum_{i=j}^{i=N-1} 1 \\
&= R_l \cdot C_l \cdot \sum_{j=0}^{j=N-1} (N-j) \\
&= R_l \cdot C_l \cdot N * (N+1)/2 \\
&= O(N^2)
\end{aligned}
$$

Wire delay growth

- Wire delay grows as $O(N^2)$ where N is length of the wire
- This is bad, as you need to (1) pipeline longer wires, causing redesigns of your FSMs + delays + area (2) there is a poor upper limit of performance for fixed cost
- Buffering is one strategy to combat wire delay growth
- With proper selection of buffer count and strength, we can make wire delay grow linearly $O(N)$.
- Linear growth is the ideal scaling factor $\rightarrow$ still only asymptotic

Optimal Buffer Placement

- Buffers (pair of inverters) can help boost signal strength along the wire
- Effect of inserting a buffer = isolation of the capacitive load along the wire
- Elmore delay must be analyzed for the buffered wire segment + capacitive load of buffer



Elmore Delay of Buffered Interconnect

$$
\begin{aligned}
T_{V_{src} \to V_{buf}} &= \sum_{j=0}^{j=\frac{N}{k}-1} T_j \\
&= R_l \cdot C_l \cdot \frac{N}{k} * (\frac{N}{k} + 1)/2 \\
T_{V_{src} \to V_{dest}} &= k \cdot (T_{V_{src} \to V_{buf}} + T_{buf}) \\
&= k \cdot (R_l \cdot C_l \cdot \frac{N}{k} * (\frac{N}{k} + 1)/2) + (k-1) \cdot T_{buf}
\end{aligned}
$$

- Assume the wire of length N segments is buffered k times *i.e.* at intervals of $\frac{N}{k}$ segments
- Delay within the buffered segment follows $O(N^2)$ scaling trend but N is now $\frac{N}{k} \to k$ is a constant, so the asymptote is still quadratic
- Each buffer has its own input capacitance C_b which contributes to charging time of T_{buf} (No need to analyze C_b separately)
- How many segments k will deliver minimum total delay $T_{V_{src} \to V_{dest}}$?

Understanding delay extremes of buffered interconnect

- Expression for delay of buffered wire:

$$T_{V_{src} \to V_{dest}} = k \cdot (R_l \cdot C_l \cdot \frac{N}{k} * (\frac{N}{k} + 1)/2) + (k-1) \cdot T_{buf}$$

- **Case 1**: $k = N$, provide a buffer every RC block

$$\begin{aligned} T_{V_{src} \to V_{dest}} &= N \cdot (R_l \cdot C_l \cdot \frac{N}{N} * (\frac{N}{N} + 1)/2) + (k-1) \cdot T_{buf} \\ &= N \cdot (R_l \cdot C_l) + (N-1) \cdot T_{buf} \end{aligned}$$

- **Case 2**: $k = 1$, no buffer at all, *i.e.* unbuffered

$$\begin{aligned} T_{V_{src} \to V_{dest}} &= 1 \cdot (R_l \cdot C_l \cdot \frac{N}{1} * (\frac{N}{1} + 1)/2) + (1-1) \cdot T_{buf} \\ &= (R_l \cdot C_l \cdot N \cdot (N+1)/2) \end{aligned}$$

Choosing the best k

- Expression for delay of buffered wire:

$$T_{V_{src} \to V_{dest}} = (R_l \cdot C_l \cdot (\frac{N^2}{k} + 1)/2) + (k-1) \cdot T_{buf}$$

- **Case 3**: T is minimized when $\frac{dT}{dk} = 0$

$$\frac{dT}{dk} = -(R_l \cdot C_l \cdot (\frac{N^2}{k^2})/2) + T_{buf}$$

$$0 = -(R_l \cdot C_l \cdot (\frac{N^2}{k^2})/2) + T_{buf}$$

$$R_l \cdot C_l \cdot (\frac{N^2}{k^2})/2 = T_{buf}$$

$$R_l \cdot C_l \cdot (\frac{N^2}{T_{buf}})/2 = k^2$$

$$k = N \cdot \sqrt{R_l \cdot C_l/2 \cdot T_{buf}}$$

Delay at best k

- Expression for delay of buffered wire: $T_{V_{src} \to V_{dest}} \approx (R_l \cdot C_l \cdot (\frac{N^2}{k})/2) + (k) \cdot T_{buf}$
- Substituting k

$$
\begin{aligned}
T_{V_{src} \to V_{dest}} &\approx (R_l \cdot C_l \cdot (\frac{N^2}{N \cdot \sqrt{R_l \cdot C_l/2 \cdot T_{buf}}} + 1)/2) + \\
&(N \cdot \sqrt{R_l \cdot C_l/2 \cdot T_{buf}}) \cdot T_{buf} \\
&\approx (R_l \cdot C_l \cdot (\frac{N}{\sqrt{R_l \cdot C_l/2 \cdot T_{buf}}})/2) + \\
&(N \cdot \sqrt{R_l \cdot C_l \cdot T_{buf}/2}) \\
&\approx N \cdot (\sqrt{R_l \cdot C_l \cdot T_{buf}/2}) + \\
&(N \cdot \sqrt{R_l \cdot C_l \cdot T_{buf}/2})
\end{aligned}
$$

- Delay is minimized when delay through RC tree = delay through buffers!

Class Wrapup

- Elmore Delay Model replaces the need for solving complex differential equations to compute time
- Represent nets are **trees** of distributed RC elements
- STA (Static Timing Analysis) must include Elmore delay calculations for accurate assesment of circuit performance
- Buffering of long wires uses Elmore delay-driven analysis