ELMORE DELAY

ESTIMATE OF $\sum$ FOR A RC TREE NETWORK

$$\sum c_i = \sum c_k r_{ik}$$

WHERE $i$ IS NODE OF INTEREST,

$c_k$ IS CAP AT NODE $k$

$r_{ik}$ IS SUM OF ALL RESISTANCE IN COMMON FROM SOURCE TO NODE $i$ + SOURCE TO NODE $k$

EXAMPLE

![Diagram of RC network with nodes and components identified](image)
For Node 1 (Delay from input to Node 1)

\[ T_1 = R_1 C_1 + R_1 C_2 + R_1 C_3 \]

(Delay for Node 2)

Node 2 \implies T_2 = C_1 R_1 + C_2 (R_1 + R_2) + C_3 R_1

Node 3 \implies T_3 = C_1 R_1 + C_2 R_1 + C_3 (R_1 + R_3)

(Delay for Node 3)

Common case is RC ladder

\[ T_N = C_1 R_1 + C_2 (R_1 + R_2) + C_3 (R_1 + R_2 + R_3) + \ldots + C_N (R_1 + R_2 + R_3 + \ldots + R_N) \]
Elmore delay is reasonably accurate

Even when Elmore delay is not accurate, it is almost always useful for optimization in that reducing Elmore delay will almost always reduce true delay.
Example (pg 163)

Driving "h" NAND gates in parallel

\[ A \rightarrow \frac{1}{2} \rightarrow \frac{1}{6C} \rightarrow h \times 4C \]

\[ B \rightarrow \frac{1}{2} \rightarrow \frac{1}{12C} \]

Let...

\[ A \rightarrow \frac{1}{2} \rightarrow \frac{1}{1} \rightarrow \frac{1}{3} R_N \]

Minimum size inverter

\[ A \rightarrow \frac{1}{2} \rightarrow \frac{1}{1} \rightarrow \frac{1}{6} R_P \]
RISING 

\[ \text{SLOWEST} \quad A = B = 1 \quad \text{THEN} \quad B = 0, \quad A = 1 \]

\[ \text{LC NEEDS TO BE CHARGED} \]

\[ \frac{1}{2C} \quad \frac{R_P}{2} \quad \frac{(6 + 4h)C}{(6 + 4h)C} \quad \text{OUT} \]

\[ \tau = (6 + 4h)C \cdot R_P + \frac{2C}{2} \cdot R_P \]

\[ \text{FASTEST} \quad A = B = 1 \quad \text{THEN} \quad A = B = 0 \]

\[ \tau = (6 + 4h)C \cdot \left( \frac{R_P}{2} \right) \]
**Falling** \( t_{df} \)

**Slowest** \( A=1, B=0 \) **Then** \( B=1, A=1 \)

So "2C" is charged near VDD and needs to be discharged.

\[
\begin{align*}
\frac{R_N}{2} & \quad \frac{1}{(6+4h)C} \\
\frac{R_N}{2} & \quad \frac{1}{2C} \\
\frac{1}{C} & \quad \frac{1}{2C} + \frac{(6+4h)C}{R_N}
\end{align*}
\]

\[C = 2C \left(\frac{R_N}{2}\right) + (6+4h)C \cdot \frac{1}{R_N}\]

**Fastest** \( A=0, B=1 \) **Then** \( A=1, B=1 \)

So "2C" is discharged already.

\[E = (6+4h)C \cdot \frac{1}{R_N}\]

So connect critical inputs to gates closer to output (in this case "A" input).