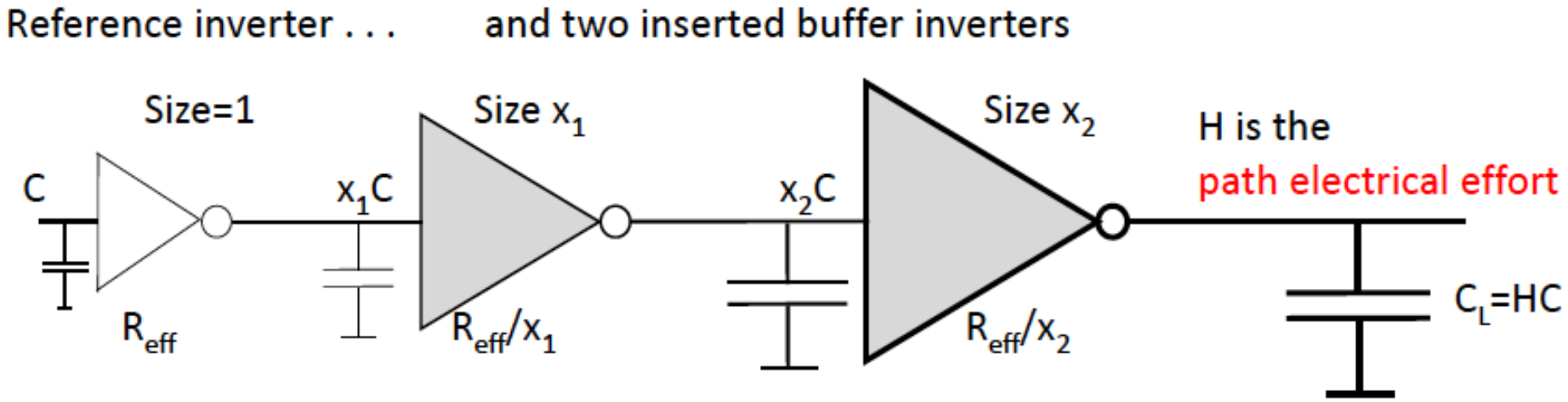
Path delay optizimization

Extending inverter delay to other types of gates

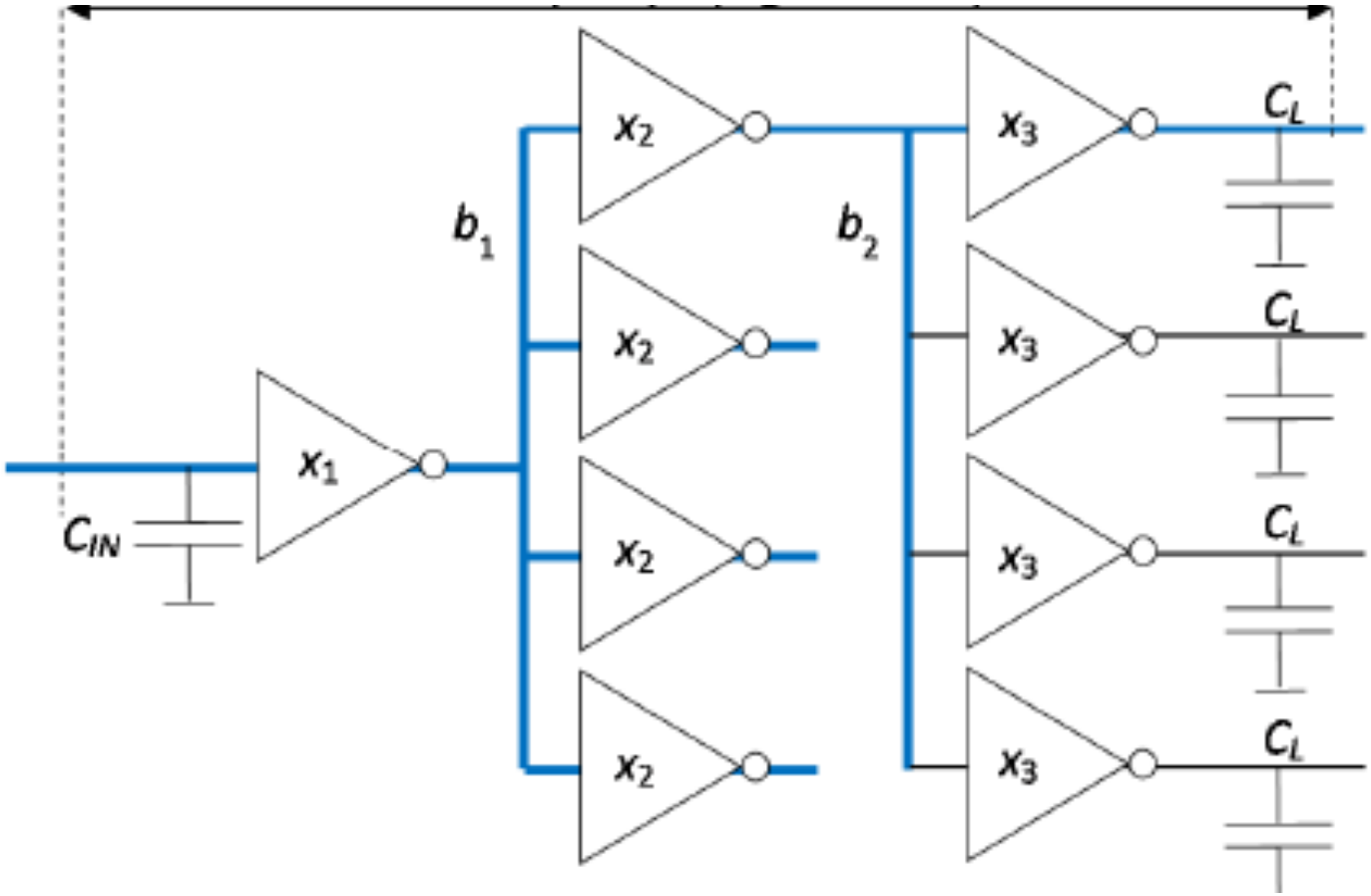
Review from lecture 4: Tapered buffer



Review from lecture 4: Tapered CMOS inverter stages

- $H = C_L/C_{IN}$: path electrical effort
- Equal stage electrical effort, h, gives shortest delay
  - $h_{opt}^N = H$, where N is number of stages
  - That is $h_{opt} = \sqrt[N]{H}$
- Normalized path delay for path is called D:
- $D = N \times h_{opt} + P$, P is sum of all parasitic delay:
- **$D = N \times h_{opt} + N \times p_{inv}$**

Review from lecture 4: CMOS inverter stages with branching



Path delay with branching

- Equal stage electrical effort gives shortest delay
- Path effort: $F = H \times B$
  - B: path branching effort
  - $B = b_1 \times b_2 \times b_3 \times \ldots b_{N-1}$ (assuming N stages)
  - $b = (c_{onpath} + c_{offpath})/c_{onpath}$
- Determine $h_{opt} = f_{opt} = \sqrt[N]{F}$
- Normalized path D:
- $D = N \times h_{opt} + P$ where P is sum of parasitic delay
- **$D = N \times h_{opt} + N \times p_{inv}$**

Path delay – when (some) gates are not inverters

- Equal stage effort still gives shortest delay
  - Now includes also logical effort: g
- Path effort: F = G×H×B
  - G: path logical effort
  - G = $g_1 \times g_2 \times g_3 \times \ldots g_N$ (assuming N stages)
  - B as defined before
- Determine f_{opt} as $\sqrt[N]{F}$.
- Normalized path delay D
- **$D = N \times f_{opt} + P$** where P is sum of parasitic delay

Approach for delay optimization= path sizing

- Given: N stages, g and p for all gates
- Calculate path effort **F** from F = G×H×B
- Calculate stage effort $\mathbf{f_{opt}}$ as $\sqrt[N]{F}$
- (Calculate path delay D = N×f_{opt} + P)
- Find gate sizes $\mathbf{X_2}$ to $\mathbf{X_N}$ starting from start or end of path.
  - Note that X_1 (input capacitance of first stage) does not change!
- Check also that f_1 /f_N (for the remaining stage) is also f_{opt} so you did not make a mistake!