Chapter 2

# Physical Structure of CMOS Integrated Circuits

何宗易

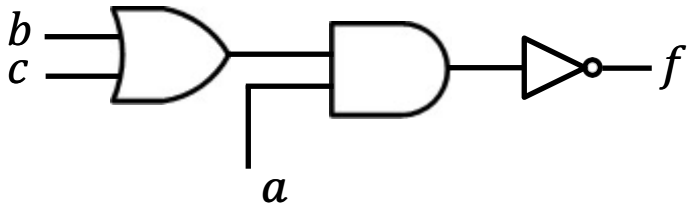
Tsung-Yi Ho
tyho@cs.nthu.edu.tw
http://theta.cs.nthu.edu.tw
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan

# References

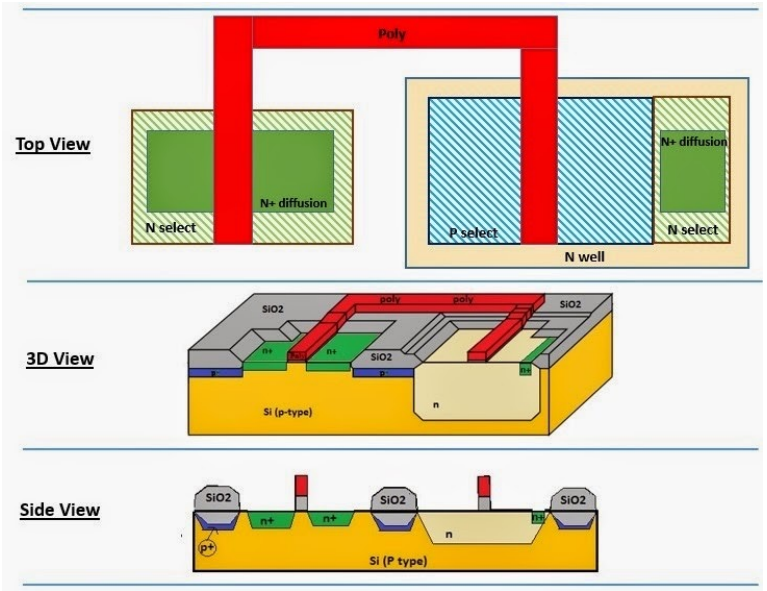

- John P. Uyemura, “Introduction to VLSI Circuits and Systems,” 2002.
  - Chapter 3
- Neil H. Weste and David M. Harris, “CMOS VLSI Design: A Circuits and Systems Perspective,” 2011.
  - Chapter 1

# Logical vs. Physical

- Logical structure




- Physical structure




Source: http://www.vlsi-expert.com/2014/11/cmos-layout-design.html

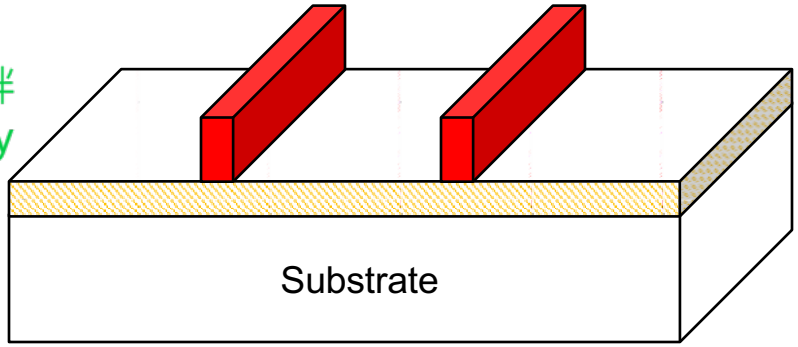
# Integrated Circuit Layers

- Semiconductor
  - Transistors (active elements)
- Conductor
  - Metal (interconnect)
    - Wire
    - Via (Via:線跟線之間換層)
- Insulator
  - Separators

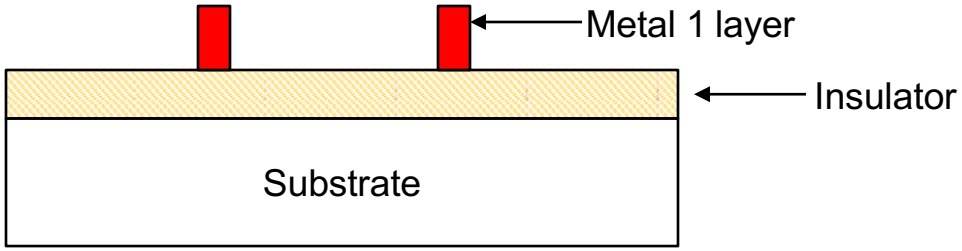
# Integrated Circuit Layers

- Silicon substrate, insulator, and two wires (3D view)

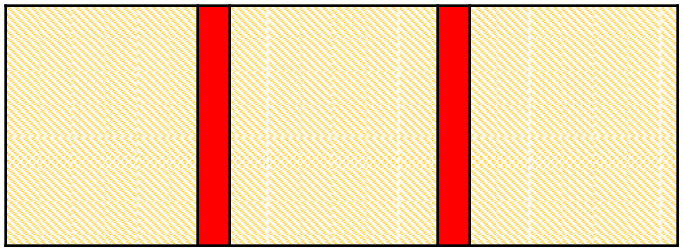
metal細長因為做扁胖的話走水平距離delay太高, 不做長的話,平行板效應又不好




- Side view




- Top view

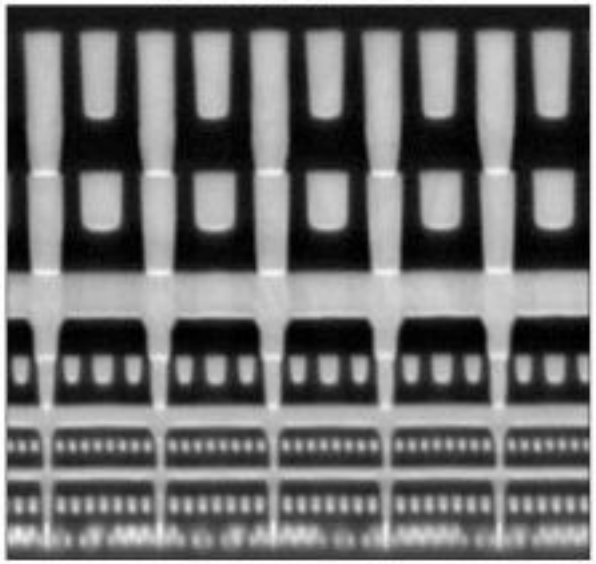


# Integrated Circuit Layers

Embargo until 8-11-14, 9 am PDT

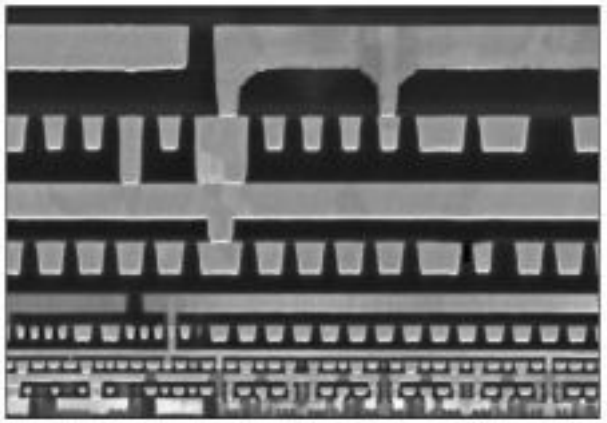
## Interconnects

22 nm Process



80 nm minimum pitch

14 nm Process



52 nm (0.65x) minimum pitch

*52 nm Interconnect Pitch Provides Better-than-normal Interconnect Scaling*

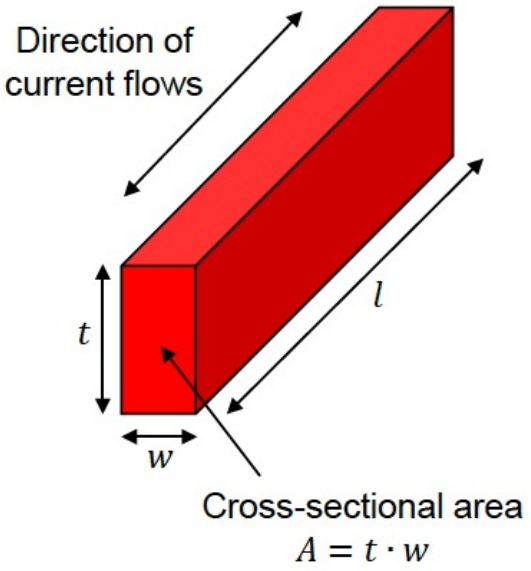
intel

# Integrated Circuit Layers

- Signal transfer speed is affected by the interconnect resistance and capacitance.
  - Resistance ↑ => Signal delay ↑
  - Capacitance ↑ => Signal delay ↑

# Integrated Circuit Layers

- Resistance
  - $R = \rho \frac{l}{A} = \frac{\rho}{t} \cdot \frac{l}{w} = R_s \cdot \frac{l}{w}$
    - $R_s$: sheet resistance (constant)
    - $\rho$: resistivity ($= \frac{1}{\sigma}$, $\sigma$: conductivity)
      - Material property (constant)
      - Unit: $\Omega \cdot m$
    - $t$: thickess (constant)
    - $w$: width (variable)
    - $l$: length (variable)

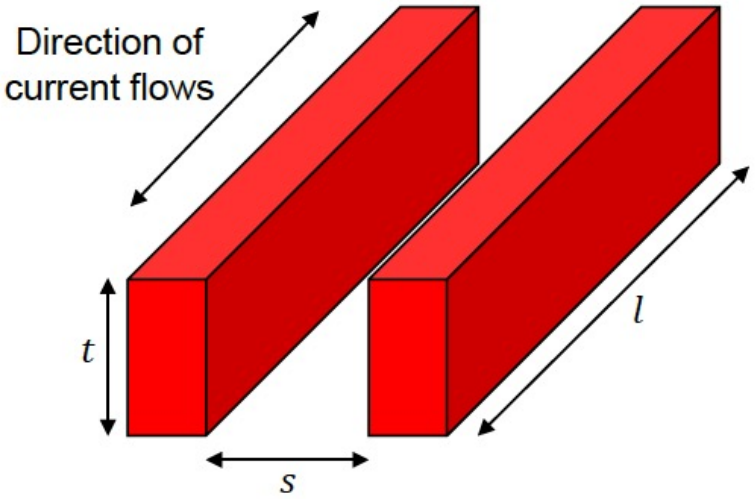



- Example
  - $\rho: 17.1 n\Omega \cdot m,\ t: 0.13 \mu m, w: 65 nm, l: 1000 \mu m$
    - $R = (17.1 \cdot 10^{-9} \Omega \cdot m) \cdot \frac{1000 \cdot 10^{-6} m}{(0.13 \cdot 10^{-6} m) \cdot (65 \cdot 10^{-9} m)} = 2023 \Omega$

# Integrated Circuit Layers

(Capacitance by Coupling 平行板效應)

- Capacitance
  - $C = \varepsilon \frac{t \cdot l}{s}$
    - $\varepsilon$: permittivity
      - Material property (constant)
      - Unit: F/m
    - $s$: distance between two conductors




- Example
  - $\varepsilon: 1.8 \cdot 10^{-11} F/m,\ t: 0.13\mu m, s: 65nm, l: 1000\mu m$
    - $C = (1.8 \cdot 10^{-11} F/m) \cdot \frac{(0.13 \cdot 10^{-6} m) \cdot (1000 \cdot 10^{-6} m)}{65 \cdot 10^{-9} m} = 3.6 \cdot 10^{-14} F = 36 fF$

=> Capacitance要提升，距離越近，overlapping的面積越高越好

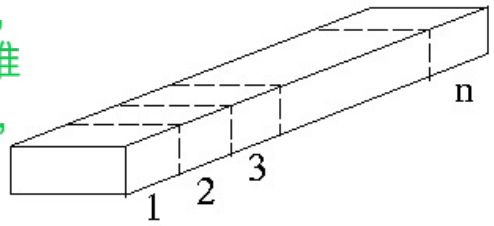
# Elmore Delay: Nonlinear Delay Model

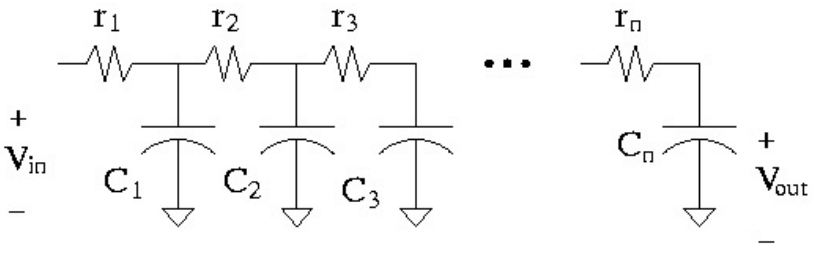
(transistor跟wire間互相干擾產生的)

- Parasitic resistance and capacitance dominate delay in deep submicron wires.
- Resistor $r_i$ must charge all downstream capacitors.
- **Elmore delay:** Delay can be approximated as sum of sections: resistance * downstream capacitance.

$$\delta = \sum_{i=1}^{n} \left( r_i \sum_{k=i}^{n} c_k \right) = \sum_{i=1}^{n} r(n-i+1)c = \frac{n(n+1)}{2} rc.$$

把wire切成n份
r_1會推動C_1,
C_2,..., r_2會推
動C_2, C_3,...,
以此類推

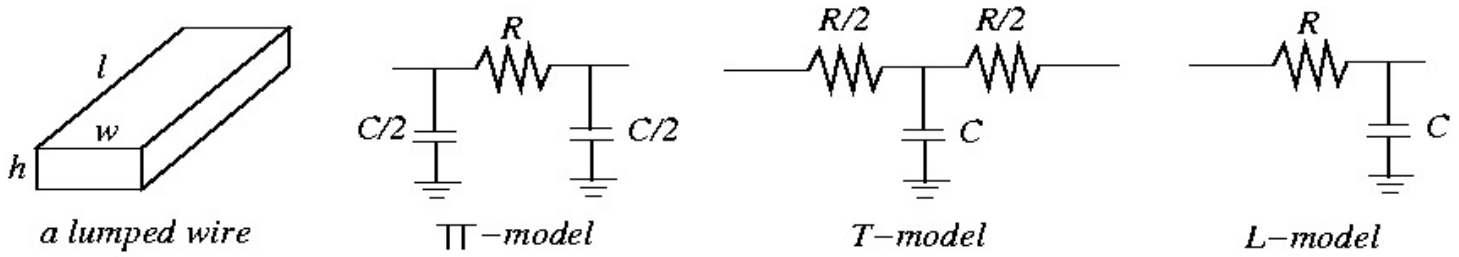



- Delay grows as **square** of wire length.
- Cannot apply to the delay with **inductance** consideration, which is important in high-performance design.

# Wire Models

- Lumped circuit approximations for distributed RC lines: π-model (most popular), *T*-model, *L*-model.

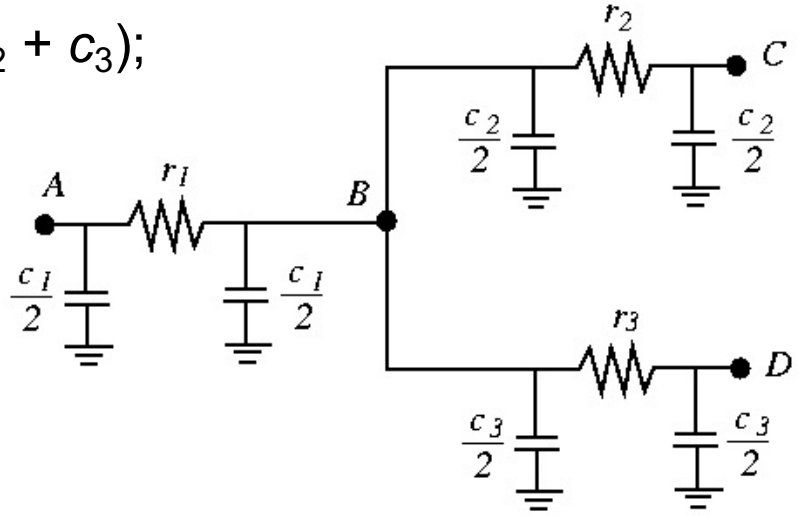



- π-model: If no capacitive loads for *C* and *D*,

  *A* to *B*: $\delta_{AB} = r_1 (c_1/2 + c_2 + c_3)$;

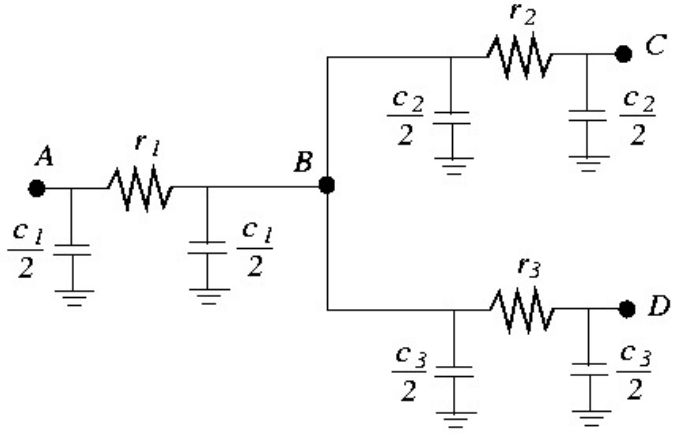
  *B* to *C*: $\delta_{BC} = r_2 (c_2/2)$;

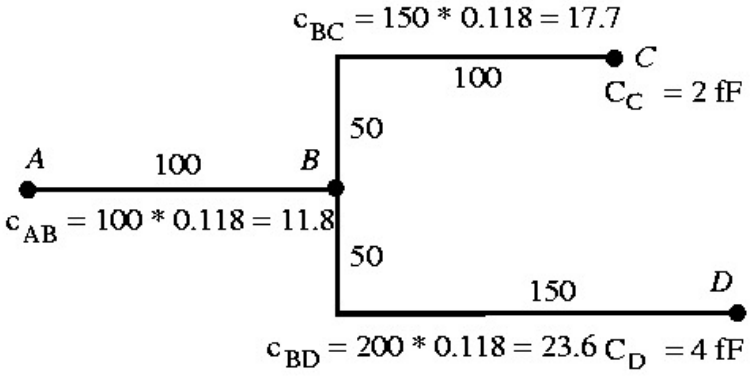
  *B* to *D*: $\delta_{BD} = r_3 (c_3/2)$.



# Example Elmore Delay Computation

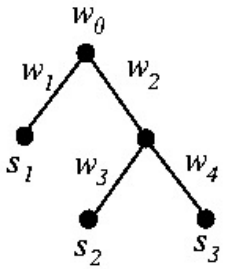
- 0.18 $\mu m$ technology: unit resistance $\hat{r}$ = 0.075 $\Omega$ / $\mu m$; unit capacitance $\hat{c}$ = 0.118 $fF/\mu m$.
  - Assume $C_C$ = 2 $fF$, $C_D$ = 4 $fF$.
  - $\delta_{BC} = r_{BC}\,(c_{BC}/2 + C_C) = 0.075 \times 150\,(17.7/2 + 2) = 120$ fs
  - $\delta_{BD} = r_{BD}\,(c_{BD}/2 + C_D) = 0.075 \times 200\,(23.6/2 + 4) = 240$ fs
  - $\delta_{AB} = r_{AB}\,(c_{AB}/2 + C_B) = 0.075 \times 100\,(11.8/2 + 17.7 + 2 + 23.6 + 4) = 400$ fs
  - Critical path delay: $\delta_{AB} + \delta_{BD}$ = 640 fs.



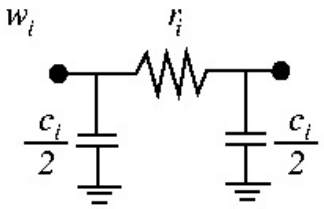



# Delay Calculation for a Clock Tree

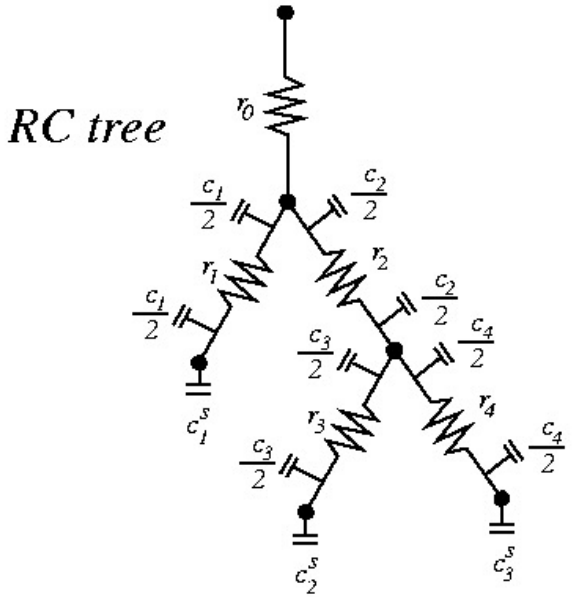
- Let $T$ be an RC tree with points $P = \{p_1, p_2, \ldots, p_n\}$, $c_i$ the capacitance of $p_i$, $r_i$ the resistance of the edge between $p_i$ and its immediate predecessor.
- The subtree capacitance at node $i$ is given as $C_i = c_i + \sum_{j \in Si} C_j$, where $S_i$ is the set of all the successors of $p_i$.
- Let $\delta(i, j)$ be the path between $p_i$ and $p_j$, excluding $p_i$ and including $p_j$.
- The delay between two nodes $i$ and $j$ is $t_{ij} = \sum_{j \in \delta(i, j)} r_j C_j$,
- $t_{03} = r_0 (c_1+c_2+c_3+c_4+c_1^s+c_2^s+c_3^s) + r_2(c_2/2 + c_3 + c_4 + c_2^s+c_3^s)+r_4(c_4/2+c_3^s)$.




clock tree




delay model



=>建一個tree , bottom up不斷從兩個leaf node找到一個tapping point往上長(每個分支都要找到tapping point)

# Exact Zero Skew Algorithm (1/2)

- Tsay, "Exact zero skew algorithm," ICCAD-91.
- To ensure the delay from the **tapping point** to leaf nodes of subtrees $T_1$ and $T_2$ being equal, it requires that

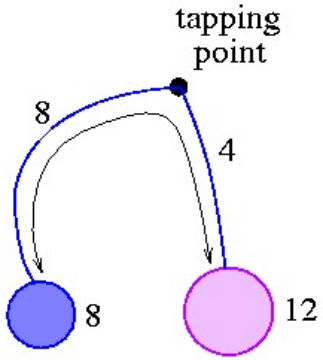
$$r_1 (c_1/2 + C_1) + t_1 = r_2 (c_2/2 + C_2) + t_2.$$

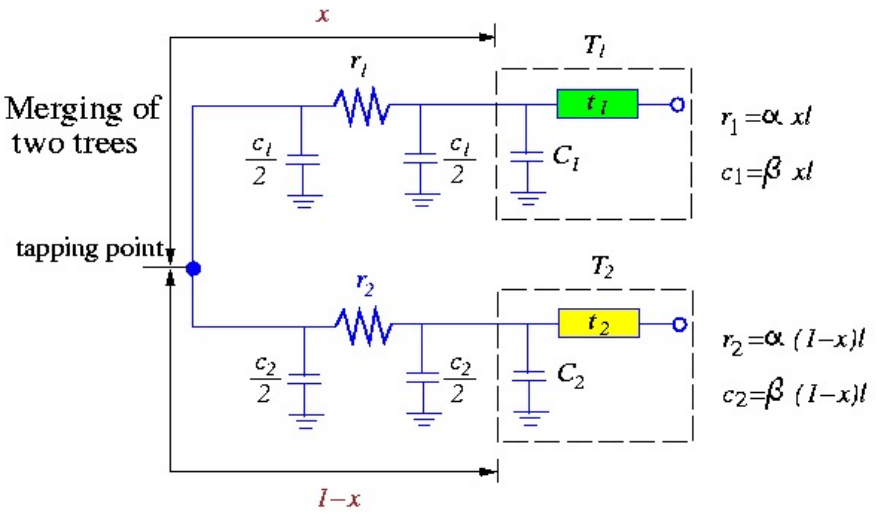
- Solving the above equation, we have

$$x = \frac{(t_2 - t_1) + \alpha l \left(C_2 + \frac{\beta l}{2}\right)}{\alpha l (\beta l + C_1 + C_2)},$$

=> clock從某個點進來後到兩個C1, C2的時間要一樣, 所以要確保從哪個點進來到的時間會一樣 (ex. clock進來到兩個D-flip flops的時間要一樣)

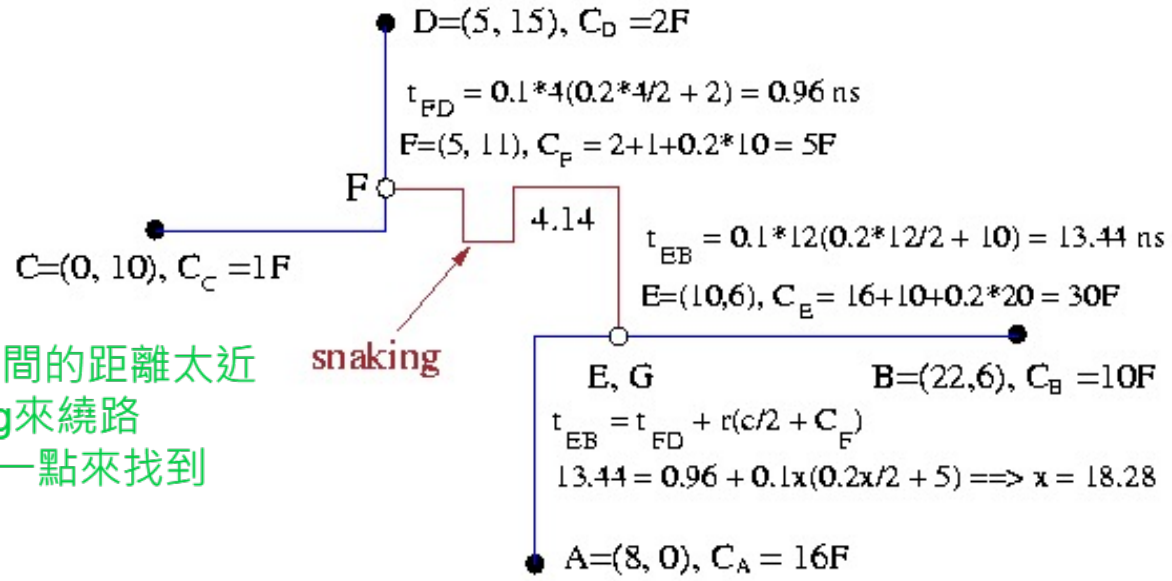
where $\alpha$ and $\beta$ are the per unit values of resistance and capacitance, $l$ the length of the interconnecting wire, $r_1 = \alpha x l$, $c_1 = \beta x l$, $r_2 = \alpha(1 - x)\, l$, $c_2 = \beta(1 - x) l$.



# Zero-Skew Computation (2/2)

- **Balance delays:** $r_1(c_1/2 + C_1) + t_1 = r_2 (c_2/2 + C_2) + t_2$.
- **Compute tapping points:** $x = \frac{(t_2 - t_1) + \alpha l \left(C_2 + \frac{\beta l}{2}\right)}{\alpha l(\beta l + C_1 + C_2)}$, $\alpha$ ($\beta$): per unit values of resistance (capacitance); $l$: length of the wire; $r_1 = \alpha x l$, $c_1 = \beta x l$; $r_2 = \alpha(1 - x) l$, $c_2 = \beta (1 - x) l$.
- If $x \notin [0, 1]$, we need **snaking** to find the tapping point.
- Exp: $\alpha$ = 0.1 Ω /unit, $\beta$ = 0.2 *F/unit* (tapping points: *E*, *F*, *G)*

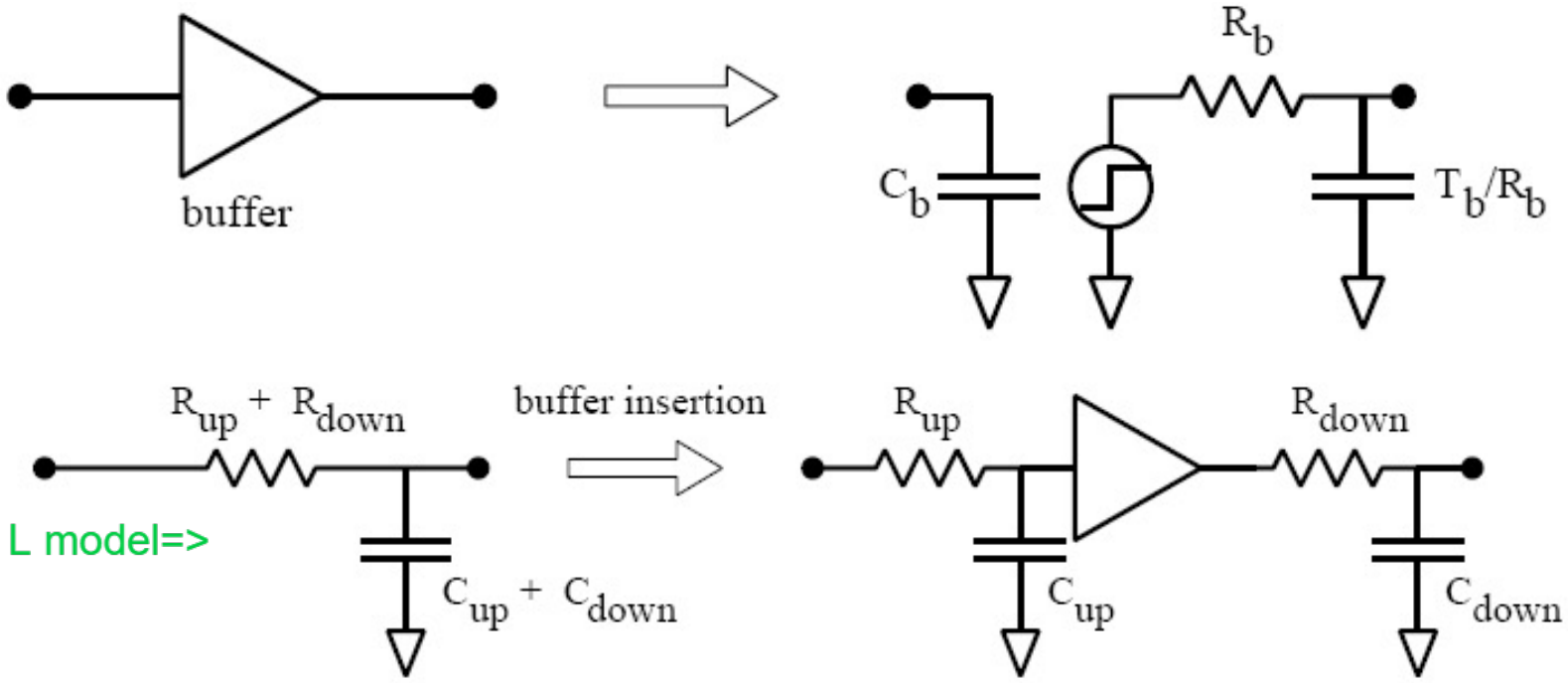



兩個點(F, EG)之間的距離太近
所以需要snaking來繞路
讓兩邊距離差大一點來找到
tapping point

# Buffer Insertion Problem

(加速用途, or 減速)

- Why Buffer Insertion?
  - Main Purpose: For Reducing Clock Phase Delay and Slew Rate




$(R_{up} + R_{down}) * (C_{up} + C_{down})$

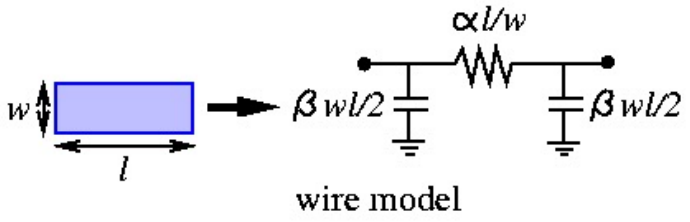
$R_{up}*(C_{up} + C_b) + T_b + (R_b + R_{down})*C_{down}$

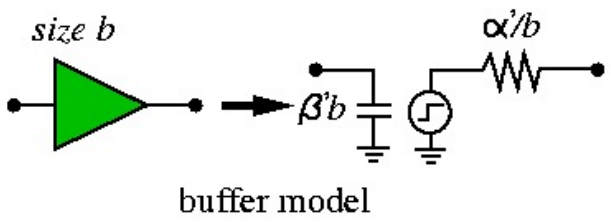
buffer 從中間切斷所以
R_up不用乘上C_down

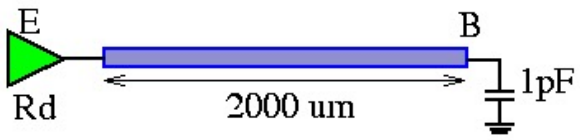
T_b: buffer自己的delay

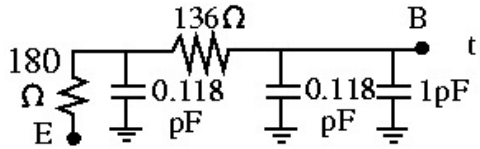
# Delay Computation for Buffered Wires

- Wire: $\alpha$ = 0.068 Ω /$\mu m$, $\beta$ = 0.118 *fF*/ $\mu m^2$; buffer: $\alpha$ ' = 180 Ω / unit size, $\beta$' = 23.4 *fF*/unit size; driver resistance $R_d$ = 180 Ω; unit-sized wire, buffer.

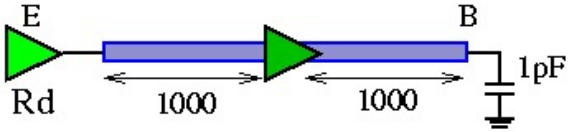


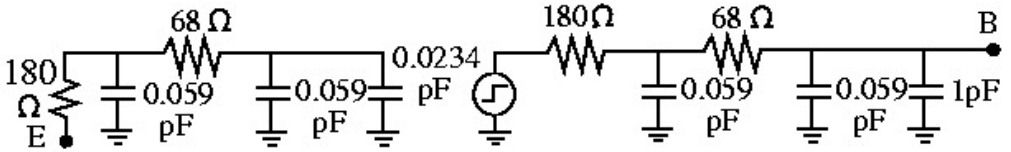



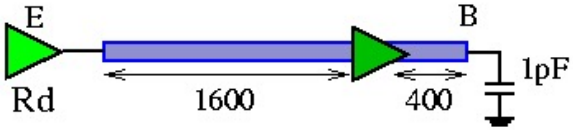



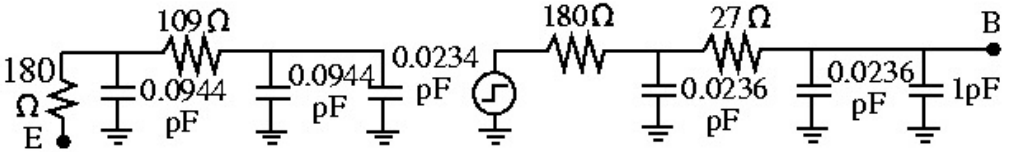

$t_{EB}$ = 180(0.118+0.118+1)+136(0.118+1)
= 375 psec




$t_{EB}$ = 180(0.059+0.059+0.0234)+68(0.059+0.0234)
+ 180(0.059+0.059+1)+68(0.059+1) = 304 psec




$t_{EB}$ = 180(0.0944+0.0944+0.0234)+109(0.0944+0.0234)
+ 180(0.0236+0.0236+1)+27(0.0236+1) = 267 psec

降低charge的capacitance => 加速

# Routing of Clock and Power Nets

- Different from other signal nets, clock and power are special routing problems
  - For clock nets, need to consider clock skew as well as delay
  - For power nets, need to consider current density (IR drop)
- => specialized routers for these nets
- Automatic tools for ASICs
- Often manually routed and optimized for microprocessors, with help from automatic tools

(IR drop=>壓降)

clock skew: clk源到每個端點的時間希望是一樣的
如果一樣, skew = 0
如果不一樣, skew = 最大delay - 最小delay

# Clock Introduction

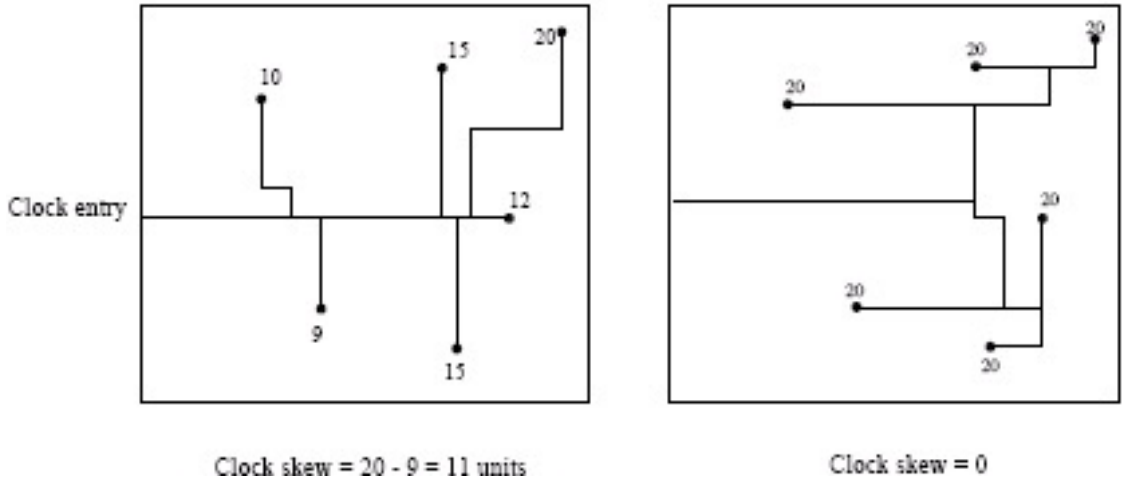
- For synchronized designs, data transfer between functional elements are synchronized by clock signals
- Clock signal are generated externally (ex: by PLL)
- Clock period equation

$$clock\ period \geq t_d + t_{skew} + t_{su} + t_{ds}$$

$t_d$: Longest path through combinational logic
$t_{skew}$: Clock skew
$t_{su}$: Setup time of the synchronizing elements
$t_{ds}$: Propagation delay within the synchronizing element

# Clock Skew

- Clock skew is the maximum difference in the arrival time of a clock signal at two different components
- Clock skew forces designers to use a large time period between clock pulses
  - This makes the system slower
- Hence, in addition to other objectives, clock skew should be minimized during clock routing




Clock skew = 20 - 9 = 11 units

Clock skew = 0

clock通常被最遠的dominate

# Clock Design Problem

- What are the main concerns for clock design?
- Skew
  - No. 1 concern for clock networks
  - For increased clock frequency, skew may contribute over 10% of the system cycle time
- Power
  - Very important, as clock is a major power consumer!
  - It switches at every clock cycle!
- Noise
  - Clock is often a very strong aggressor (一直switch所以很容易干擾或受干擾)
  - May need shielding
- Delay
  - Not really important
  - But slew rate is important (sharp transition)

# The Clock Routing Problem

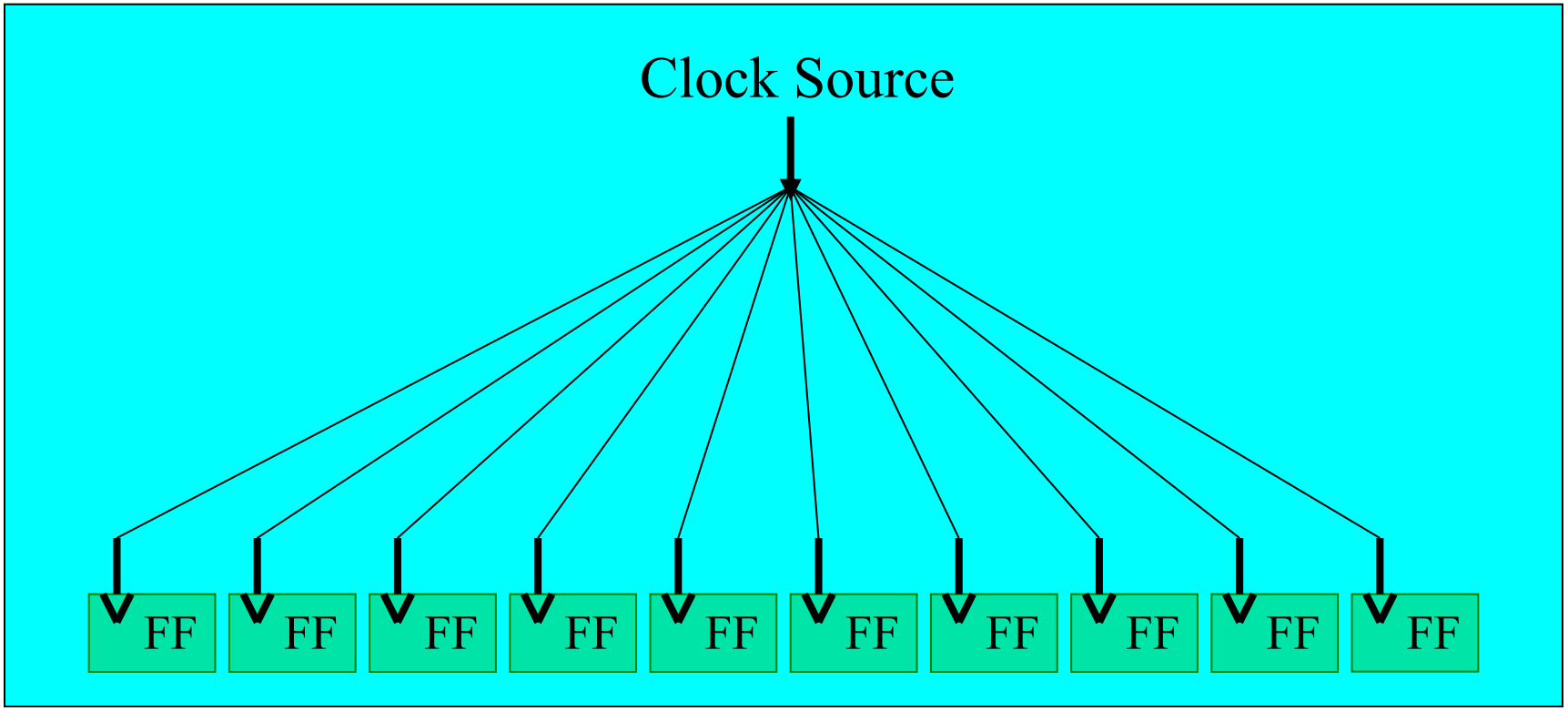
- Given a source and *n* sinks
- Connect all sinks to the source by an interconnect network (tree or non-tree) so as to minimize:
  - Clock Skew = $\max_{i,j} |t_i - t_j|$
  - Delay = $\max_i t_i$
  - Total wirelength
  - Noise and coupling effect

# Clock Design Considerations

- Clock signal is global in nature, clock nets are usually very big
  - Significant interconnect capacitance and resistance
- So what are the techniques?
  - Routing
    - Clock tree versus clock mesh (non-tree or grid)
    - Balance skew and total wire length
  - Buffer insertion
    - Clock buffers to reduce clock skew, delay, and distortion in waveform
  - Wire sizing
    - To further tune the clock tree/mesh

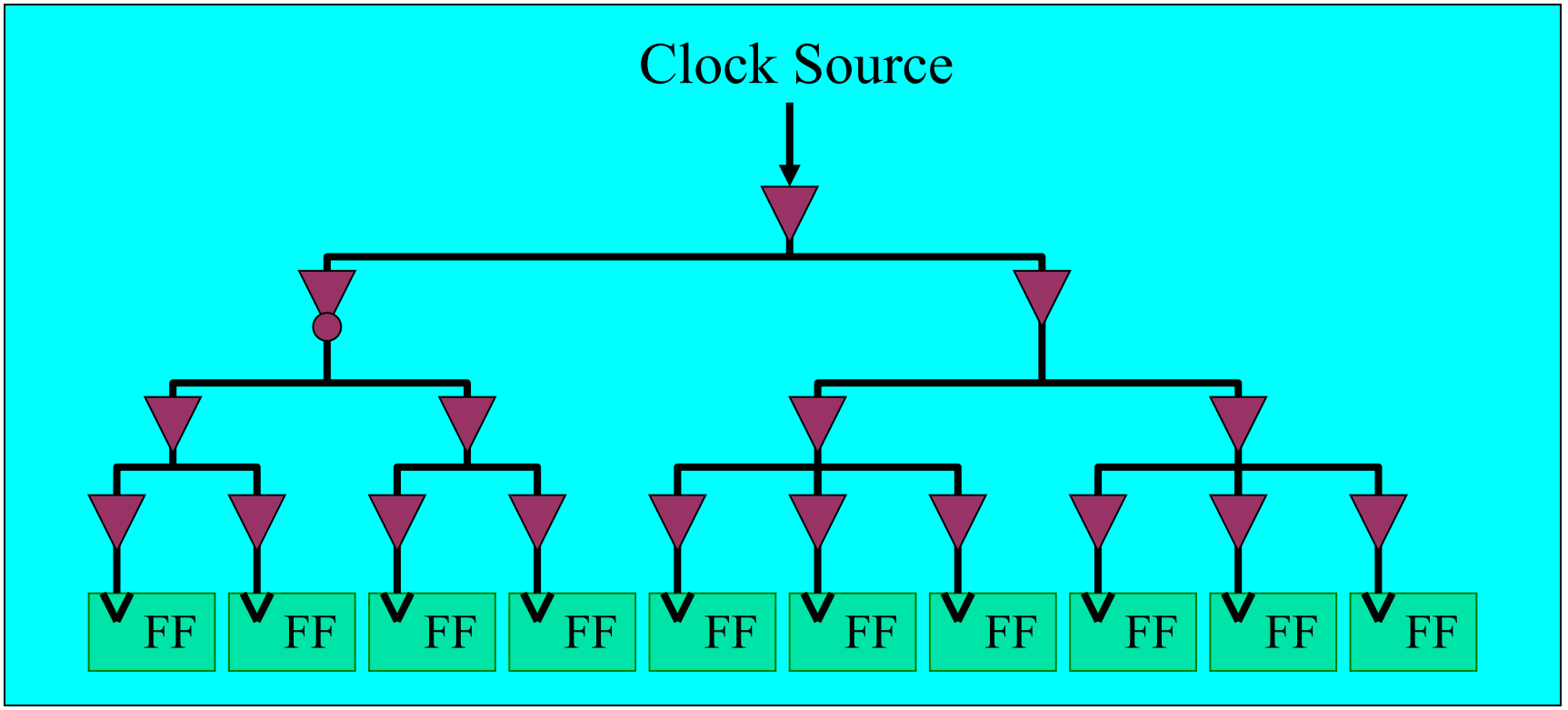
# Clock Trees

- A path from the clock source to clock sinks

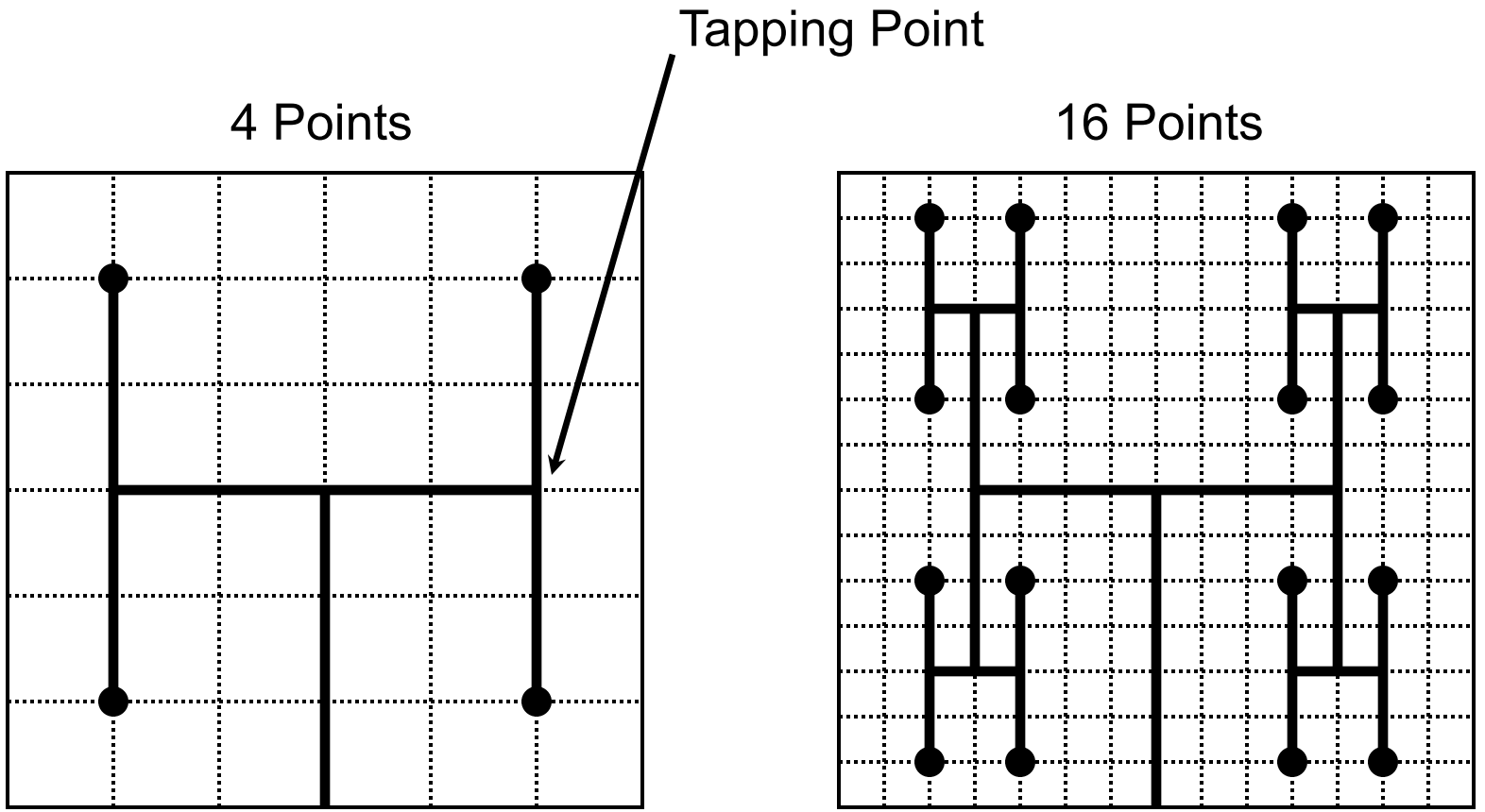


# Clock Trees

- A path from the clock source to clock sinks



# H-Tree Clock Routing



# H-tree Algorithm

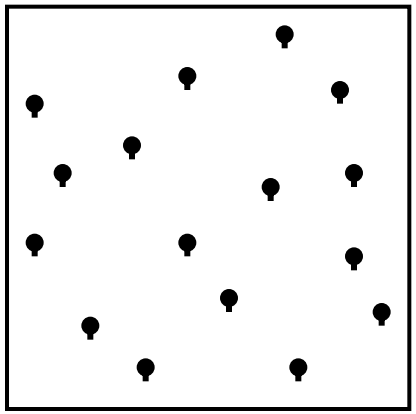
- Minimize skew by making interconnections to subunits equal in length
  - Regular pattern (只能用在規則情形)
  - The skew is 0 assuming delay is directly proportional to wirelength
    - Is this always the case???
- Can be used when terminals are evenly distributed
  - However, this is never the case in practice (due to blockage, and so on)
  - So strict (pure) H-trees are rarely used
  - However, still popular for top-level clock network design
  - Cons: too costly to be used everywhere

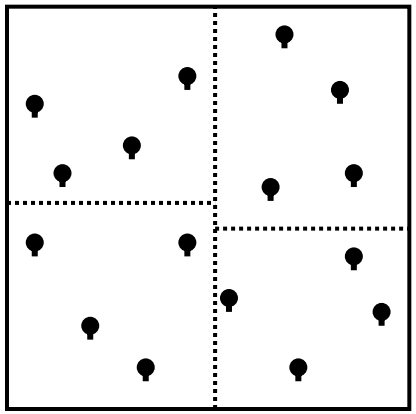
Can you think of another shape if non-rectilinear wires are allowed?

# Method of Means and Medians (MMM)

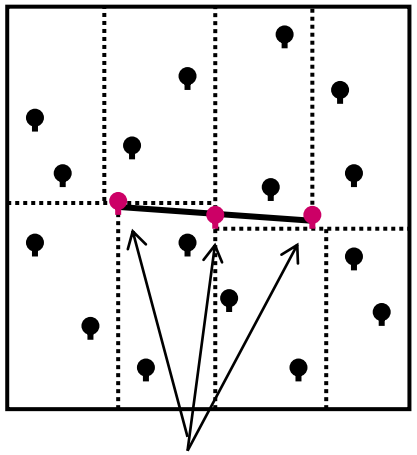
- Applicable when the clock terminals are arbitrarily arranged
- Follows a strategy very similar to H-Tree
  - Recursively partition the terminals into two sets of equal size (median)
  - Then, connect the center of mass of the whole circuit to the centers of mass of the two sub-circuits (mean)
- Clock skew is only minimized heuristically
  - The resulting tree may not have zero-skew

# An Example of MMM

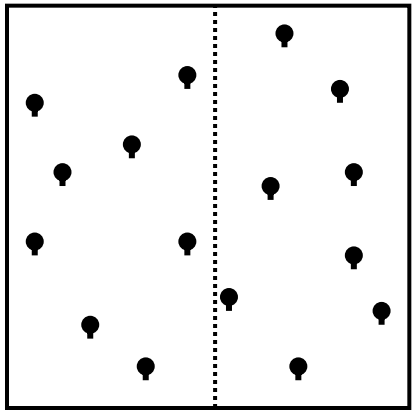




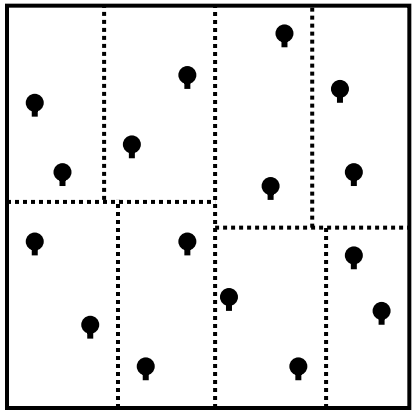
=>recursive去partition, 每次切都算質心

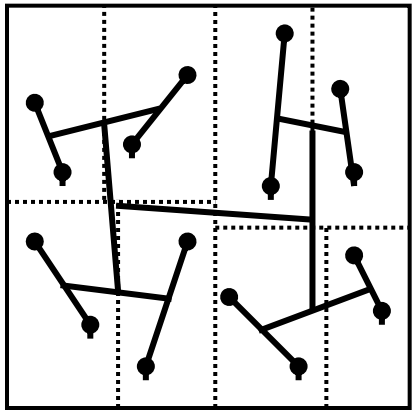


centers of mass



=>一開始第一切影響很大



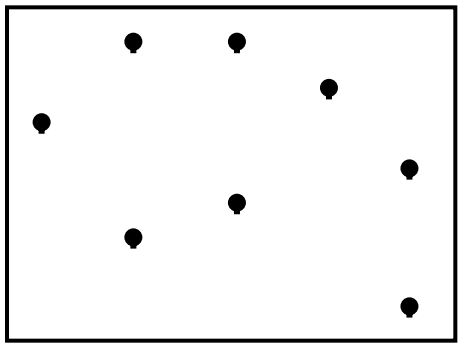


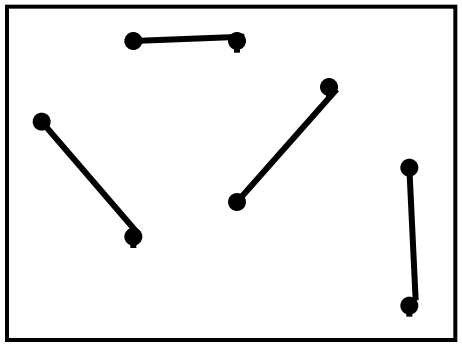
同區兩個點連起來
再跟其他區連

# Geometric Matching Algorithm (GMA)

- MMM is a top-down algorithm, but GMA is a bottom-up algorithm
- Geometric matching of *n* endpoints:
  - Construct a set of *n*/2 line segments connecting *n* endpoints pairwise
  - No two line segments share an endpoint
  - The cost is the sum of the edge lengths
- The basic idea is to find a minimum cost geometric matching recursively

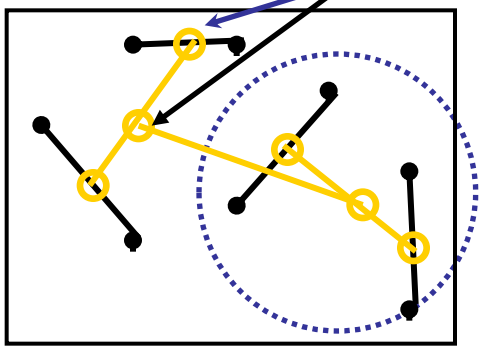
# An Example of GMA



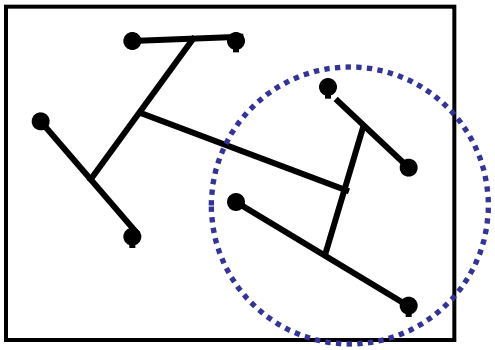


Apply geometric matching recursively

對每個點找最近且沒被連起來的




H-flipping



Post-processing

Can give zero skew clock tree

# An Exact Zero Skew Algorithm

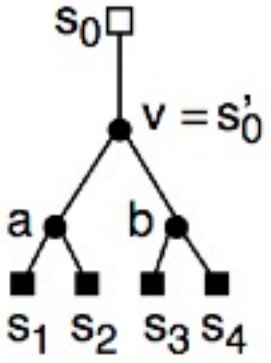
- ICCAD 1991 and TCAD 1993, Ren-Song Tsay
- A classic paper to manage clock skew
- Use Elmore delay model to compute delay
- Guarantee zero skew
  - Can easily to extended for zero skew or bounded skew
  - Can you think of a method to do it?
- Try to minimize wire length, but not done very well
  - Lots of follow up works to minimize total wire length while maintaining zero skew
  - DME and its extensions

# Deferred Merge Embedding

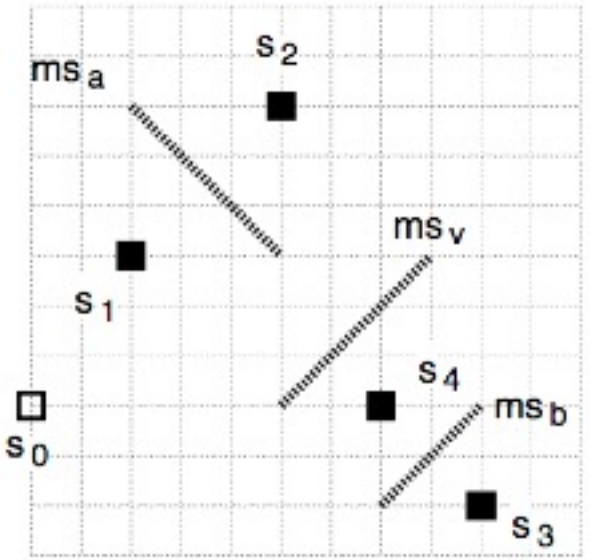
- As its name implies, DME defers the merging as late as possible, to make sure minimal wire length cost for merging
- Independently proposed by several groups
  - Edahiro, NEC Res Dev, 1991
  - Chao et al, DAC'92
  - Boese and Kahng, ASIC'92
- DME needs an *abstract routing topology* as the input
- It has a bottom-up phase followed by a top-down process

# DME Zero-Skew Algorithm (1/2)

- Use *Deferred-Merge Embedding* (**DME**) algorithm
- Given a set of clock pins and an abstract topology G
  1. Bottom-up: Get the possible locations of internal nodes. (building *merging segments*)
  2. Top-down: Determine the location points of internal nodes.

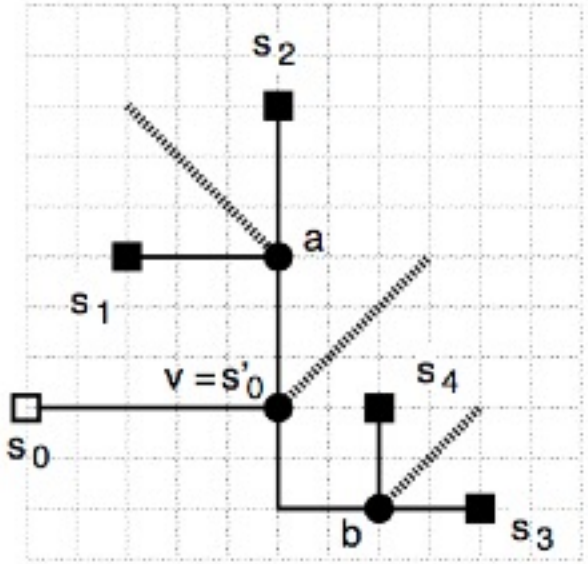


(a) Topology



(b) Bottom-up merging phase

=>把可能的中點先用線段表示

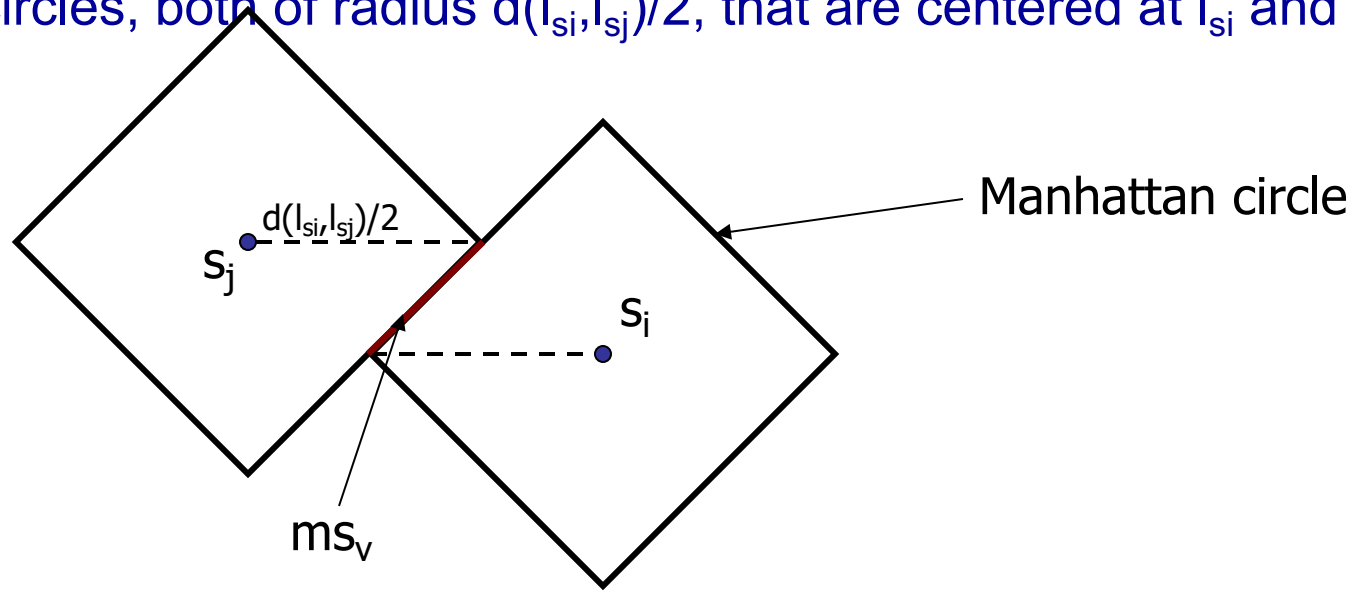


Cost = 22, Skew = 0

=>再用top-down來決定選線上哪一點

# DME Zero-Skew Algorithm(2/2)

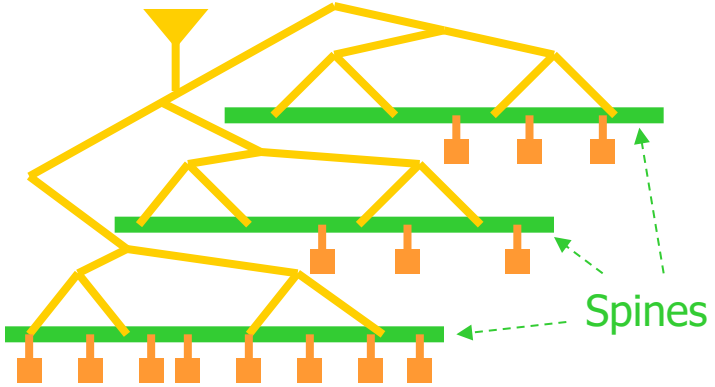
- Merging segment:
  - Merging segment of v, denoted as $ms_v$
  - With slope +1 or -1. Since the Manhattan circle is a square rotated 45°
- Example:
  - A merging segment which is the intersection of two Manhattan circles, both of radius $d(l_{si},l_{sj})/2$, that are centered at $l_{si}$ and $l_{sj}$.



# Some Thoughts/Trend

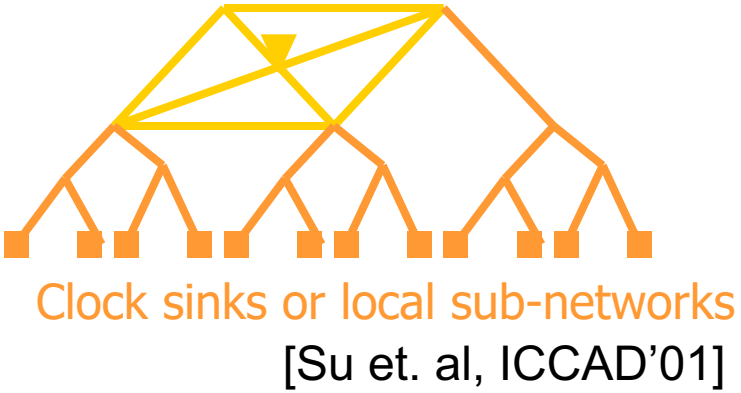
- Clock skew scheduling together with clock tree synthesis
  - Schedule the timing slack of a circuit to the individual registers for optimal performance and as a second criteria to increase the robustness of the implementation w.r.t. process variation
- Variability is a major nanometer concern
- Non-tree clock networks for variation-tolerance
  - How to analyze it?
  - The task is to investigate a combined optimization such that clock skew variability is reduced with minimum wirelength penalty

# Non-tree: Spine & Mesh

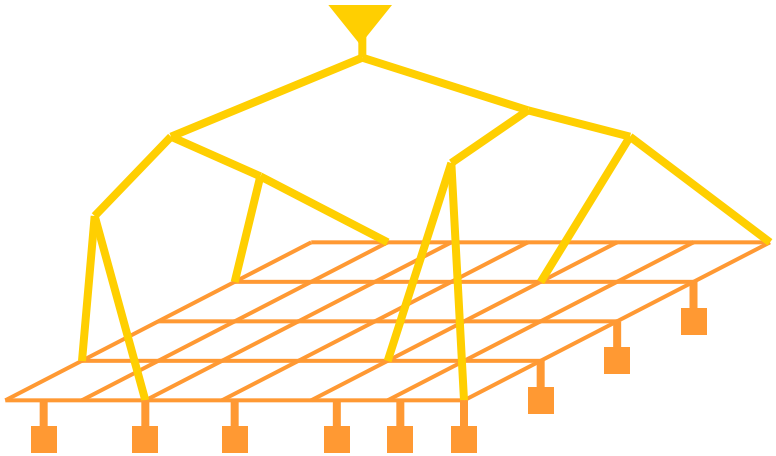



Applied in *Pentium* processor

[Kurd et. al. JSSC'01]




[Su et. al, ICCAD'01]



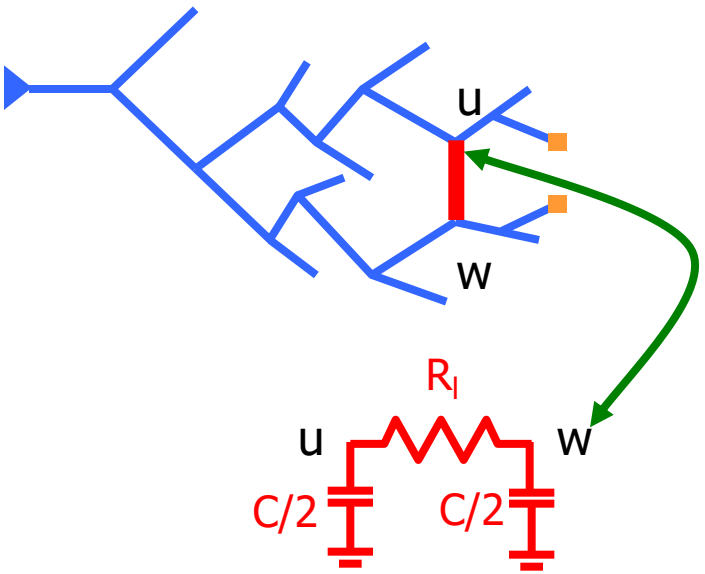

[Restle et. al, JSSC'01]

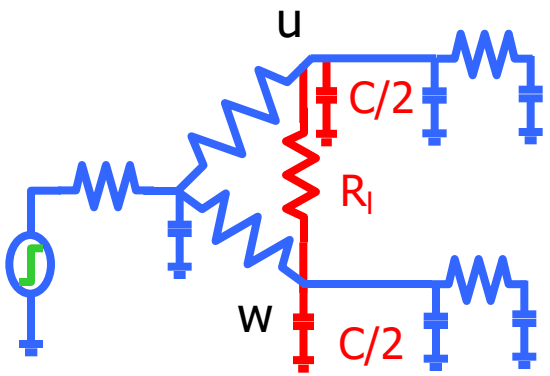
Applied in IBM microprocessor

- Very effective, huge wire

# Non-tree: Link Perspective

- Non-tree = tree + links
- How to select link pairs is the key problem
- Link = link_capacitors + link_resistor
- Key issue: find the best links that can help the skew variation reduction the most!

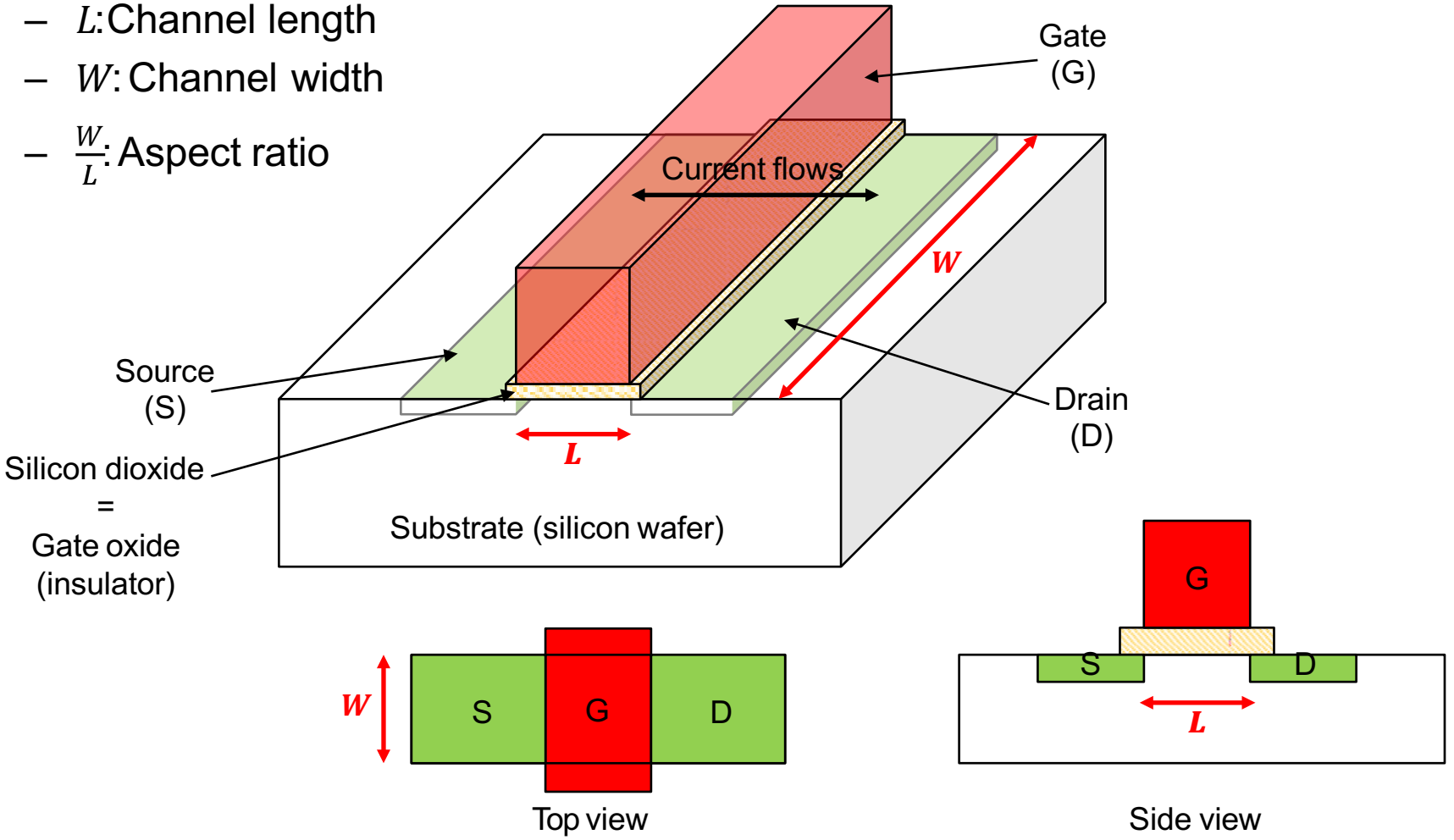



[Rajaram et al, DAC'04]

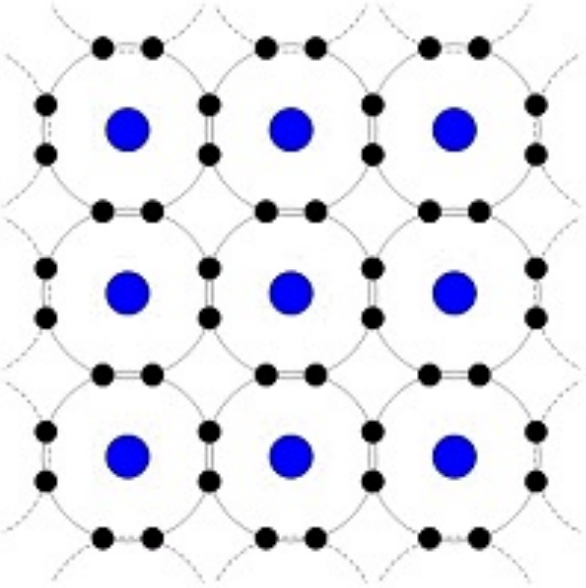
# MOSFETs – Physical Shape

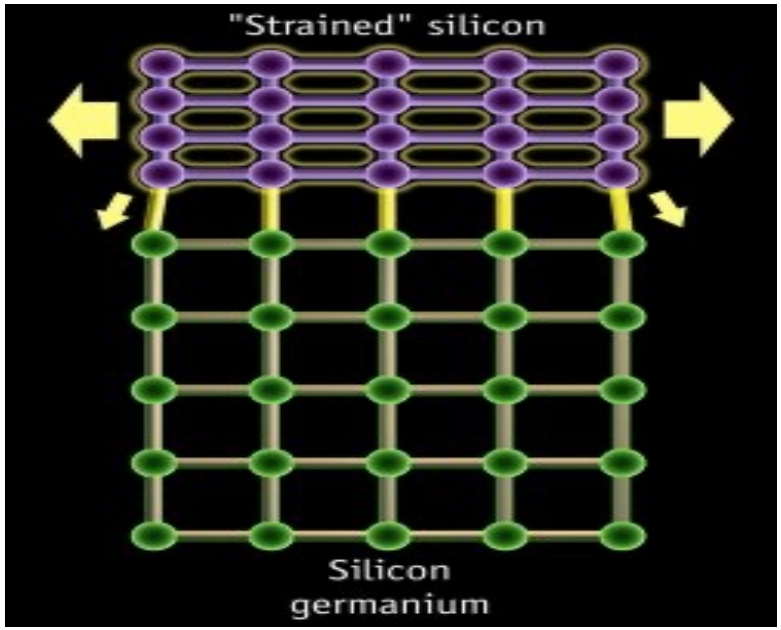
- What a MOSFET looks like at the physical level
  - $L$: Channel length
  - $W$: Channel width
  - $\frac{W}{L}$: Aspect ratio



# MOSFETs – Device Physics

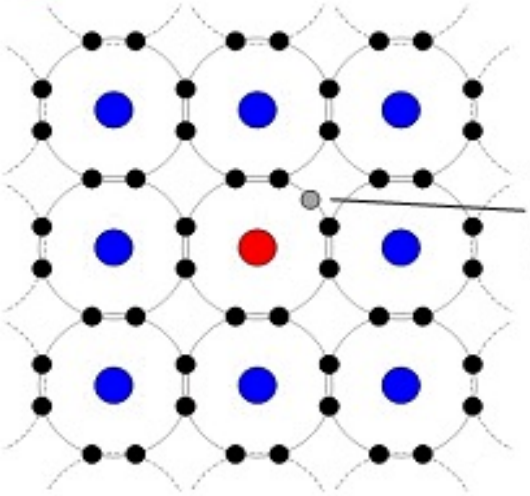
- Atomic density of a silicon crystal
  - $N_{Si} \approx 5 \times 10^{22}$
- Intrinsic carrier density
  - # free electrons (due to thermal excitations)
  - $n_i \approx 1.45 \times 10^{10}/cm^3$ (at room temperature)
- Mass action law when no current flows in pure silicon
  - $n = p = n_i$
  - $np = {n_i}^2$
    - $n$: # free electrons
    - $p$: # free holes





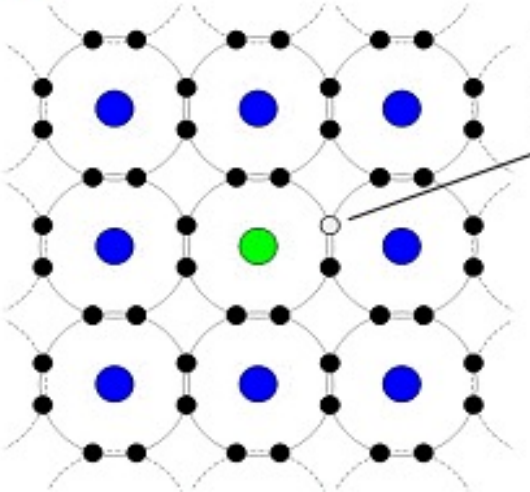

## N-Type Silicon

Phosphorous nucleus




## P-Type Silicon

Boron nucleus



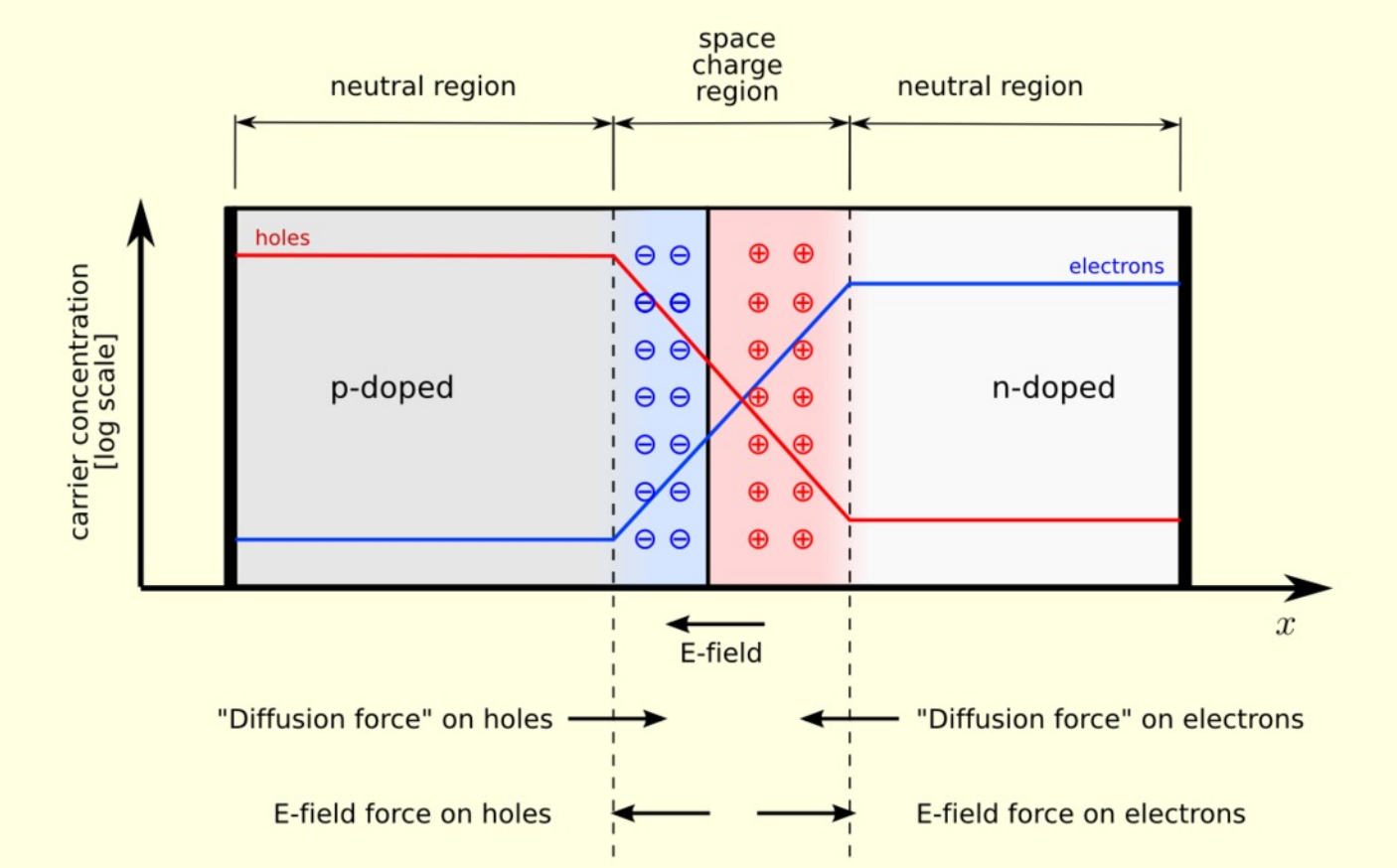
# MOSFETs – Device Physics

- Doping
  - Add impurity atoms (dopants) to enhance # electrons or # holes.
  - n-type material: if more electrons are added (donors).
    - $N_d$: # donors $(10^{16}\sim10^{19}/cm^3)$
    - # free electrons (majority carriers): $n_n \approx N_d/cm^3$
    - # holes (minority carriers): $p_n \approx \frac{{n_i}^2}{N_d}/cm^3$
    - $n_n \gg p_n$
  - p-type material: if more holes are added (acceptors).
    - $N_a$: # acceptors $(10^{14}\sim10^{19}/cm^3)$
    - # holes (majority carriers): $p_p \approx N_a/cm^3$
    - # free electrons (minority carriers): $n_p \approx \frac{{n_i}^2}{N_a}/cm^3$
    - $p_p \gg n_p$

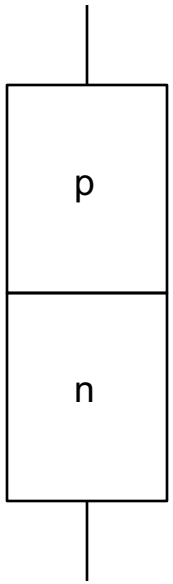
# MOSFETs – Device Physics

- Conductivity
  - $\sigma = q(\mu_n \cdot n + \mu_p \cdot p)$
    - $q$: The charge of an electron $(-1.602 \cdot 10^{-19})$
    - $\mu_n$: Electron mobility $(1360 cm^2/V \cdot s)$
    - $\mu_p$: Hole mobility $(480 cm^2/V \cdot s)$
- Intrinsic silicon
  - $\sigma \approx 4.27 \cdot 10^{-6}$
  - $\rho \approx 2.34 \cdot 10^{5}$
- Quartz glass (insulator)
  - $\rho \approx 10^{12}$
- Mobility
  - $\boldsymbol{\mu_n > \mu_p}$
- Impurity scattering
  - Adding a large number of impurity atoms reduces the mobility.

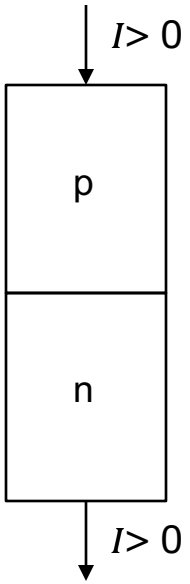
# Zero Bias


neutral region
space charge region
neutral region
holes
electrons
p-doped
n-doped
carrier concentration [log scale]
x
E-field
"Diffusion force" on holes
"Diffusion force" on electrons
E-field force on holes
E-field force on electrons

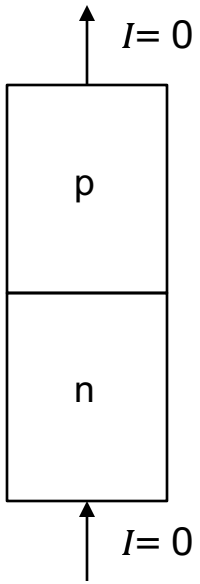
# PN Junction




pn junction

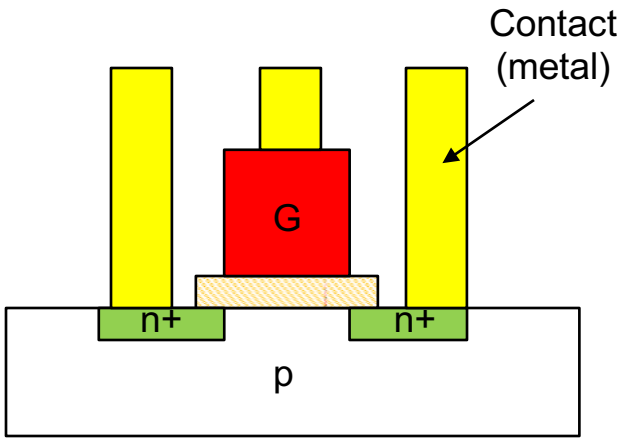



Forward current

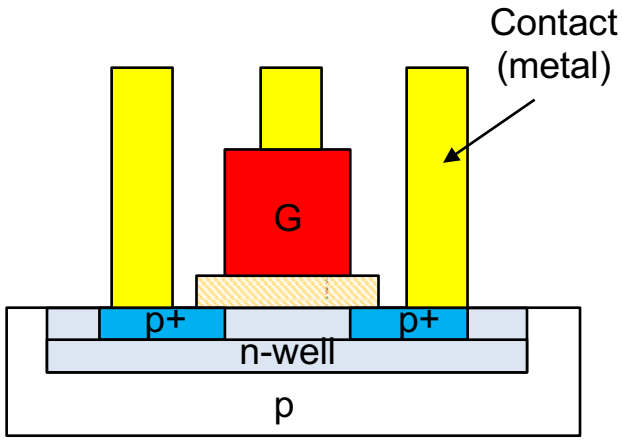



Reverse blocking

# MOSFETs




nFET




pFET

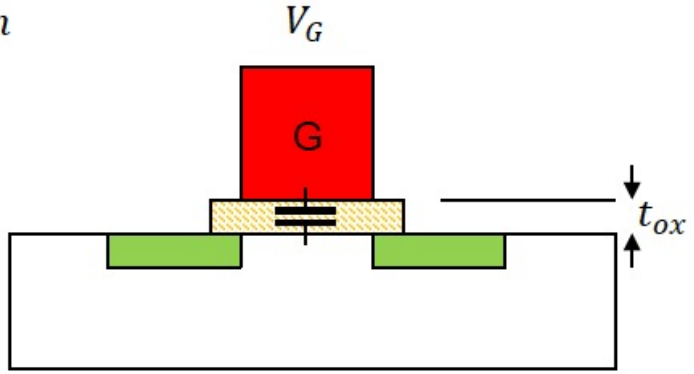
n+: heavily doped with donors

p+: heavily doped with acceptors

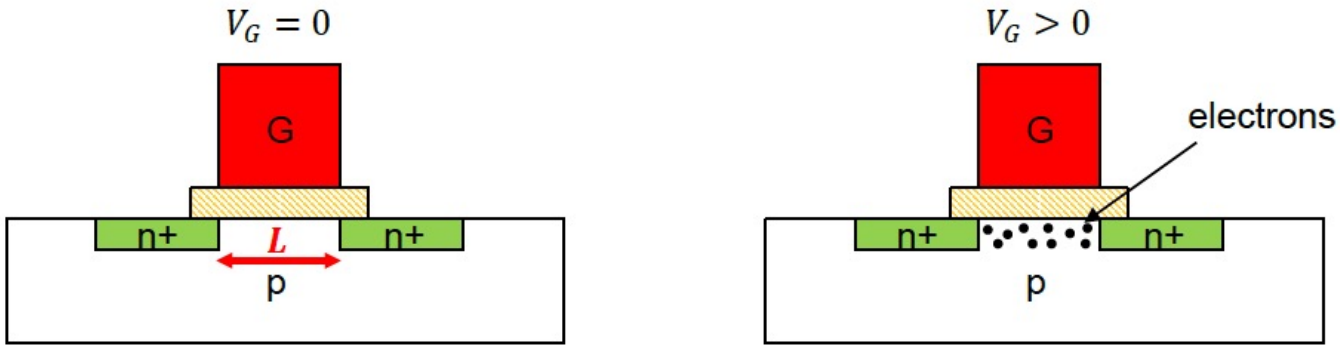
* Contacts are used to connect source/drain/gate to metal 1.

# MOSFETs – Device Physics

- $t_{ox}$: oxide thickness
  - Typically a few nm
- Gate material
  - Polysilicon (called poly)
  - Metal
- Oxide capacitance (Gate(M) – Insulator(O) – Semiconductor(S))
  - $C_G = c_{ox} \cdot A_G$
    - $c_{ox} = \frac{\varepsilon_{ox}}{t_{ox}}$: unit gate capacitance
      - $\varepsilon_{ox} \approx 3.9\varepsilon_0 = 3.9 \cdot 8.854 \cdot 10^{-12} F/m$
    - $A_G$: gate area $(= L \cdot W)$
  - Example
    - $t_{ox} = 8nm, L = 45nm, W = 70nm$
      - $C_G \approx 0.013fF$



# MOSFETs – Device Physics (nFET)

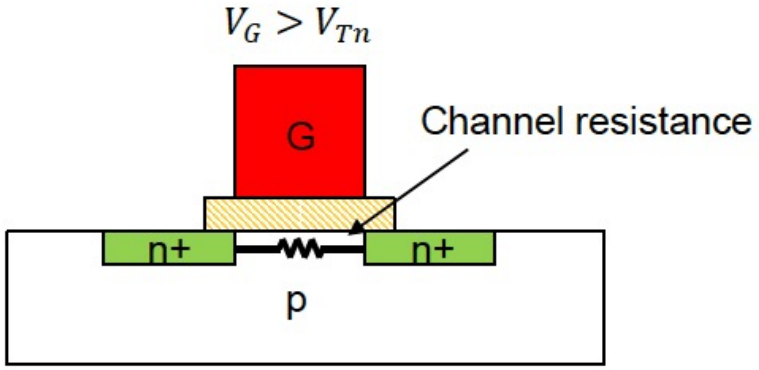



- Current
  - Channel charge: $Q_c = -C_G(V_G - V_{Tn})$
    - No charge forms until $V_G$ reaches $V_{Tn}$.
  - Current flowing the channel: $I = \frac{|Q_c|}{\tau_t}$
    - $\tau_t = \frac{L}{v}$: channel transit time (the average time needed for an electron to move from S to D).
    - $v = \mu_n \cdot E = \mu_n \cdot \frac{V_{DS}}{L}$
  - $\boldsymbol{I \approx \mu_n \cdot C_{ox} \cdot \left(\frac{W}{L}\right) \cdot (V_G - V_{Tn}) \cdot V_{DS}}$

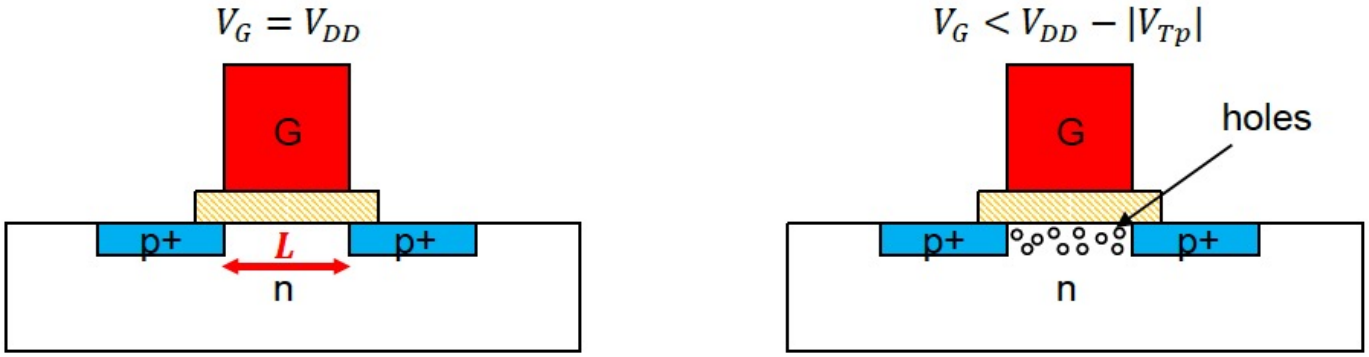
# MOSFETs – Device Physics (nFET)

- Current through the channel
  - $I \approx \mu_n \cdot c_{ox} \cdot \left(\frac{W}{L}\right) \cdot (V_G - V_{Tn}) \cdot V_{DS} = \beta_n \cdot (V_G - V_{Tn}) \cdot V_{DS}$
    - $\beta_n = \mu_n \cdot c_{ox} \cdot \left(\frac{W}{L}\right)$: device transconductance
    - $\mu_n, c_{ox}, V_{Tn}$: constants
    - $L, W$: variables (designers can decide)
    - $V_G, V_{DS}$: variables (but either $0$ or $V_{DD}$)

- Channel resistance
  - $R_n = \frac{V_{DS}}{I} = \frac{1}{\beta_n \cdot (V_G - V_{Tn})}$



# MOSFETs – Device Physics (pFET)

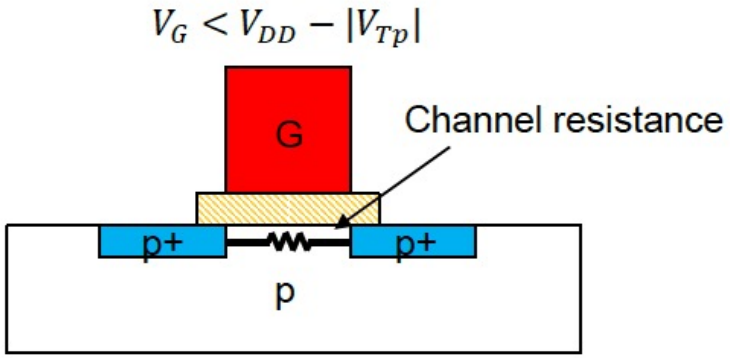



- Current
  - Channel charge: $Q_c = C_G(V_G - |V_{Tp}|)$
    - No charge forms until $V_G$ reaches $V_{DD} - |V_{Tp}|$.
  - Current flowing the channel: $I = \frac{|Q_c|}{\tau_t}$
    - $\tau_t = \frac{L}{v}$: channel transit time (the average time needed for an electron to move from D to S).
    - $v = \mu_p \cdot E = \mu_p \cdot \frac{V_{SD}}{L}$
  - $\boldsymbol{I \approx \mu_p \cdot c_{ox} \cdot \left(\frac{W}{L}\right) \cdot (V_G - |V_{Tp}|) \cdot V_{SD}}$

# MOSFETs – Device Physics

- Current through the channel
  - $I \approx \mu_p \cdot c_{ox} \cdot \left(\frac{W}{L}\right) \cdot \left(V_G - |V_{Tp}|\right) \cdot V_{SD} = \beta_p \cdot \left(V_G - |V_{Tp}|\right) \cdot V_{SD}$
    - $\beta_p = \mu_p \cdot c_{ox} \cdot \left(\frac{W}{L}\right)$: device transconductance
    - $\mu_p, c_{ox}, V_{Tp}$: constants
    - $L, W$: variables (designers can decide)
    - $V_G, V_{SD}$: variables (but either $0$ or $V_{DD}$)

- Channel resistance
  - $R_p = \frac{V_{SD}}{I} = \frac{1}{\beta_p \cdot (V_G - |V_{Tp}|)}$



# MOSFETs – Device Physics

- Charging the gate requires current flows.
  - $i = C_G \frac{dV_G}{dt}$
  - The transistor itself has a signal delay.
  - If $C_G$ is large, the delay goes up.

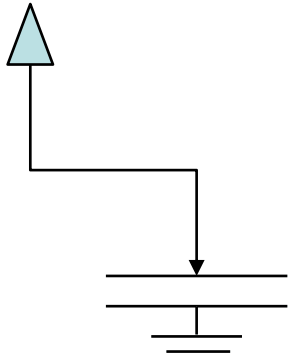
- Energy
  - $E = \int P\,dt = \int (V \cdot I) dt = \int \left( V \cdot C \frac{dV}{dt} \right) dt = \frac{1}{2} C V^2$
  - $E = \frac{1}{2} C_G {V_{DD}}^2$
  - Driving a transistor consumes energy (power dissipation).

# Driving the Gate Capacitance

- The value of the capacitance $C_G$ ($Q=C_GV_G$) determine the amount of charge necessary to charge the voltage, so a large capacitance implies a long delay.

- Both the energy stored in a capacitor and the power dissipated is $E_e = 1/2C_GVdd^2$

- Resistors dissipates power by changing it into heat.
- Charge and discharge will be $2*E_e=C_GVdd^2$



# Driving the Gate Capacitance

- Computation causes switching
- Switching delay are due to the physical characteristics of the devices and interconnect.
  - Understand the nature of switching for fast circuit
- Every switching event requires energy transfer in the circuit. This implies that power dissipation.

# Metal interconnect layer

- Modern process allow five or more metal interconnect layers to ease the problem of massive wiring in complex circuits
  - Metal layers are electrically isolated from each other.
  - A hole called contact cut is etched in the oxide.
  - Electrical contact requires contact cuts and vias

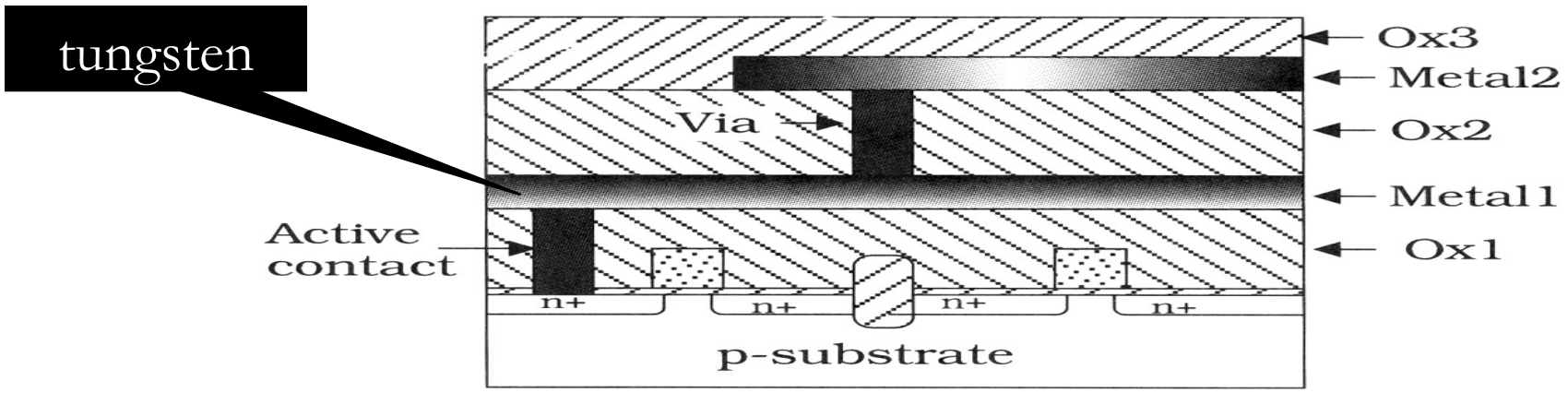
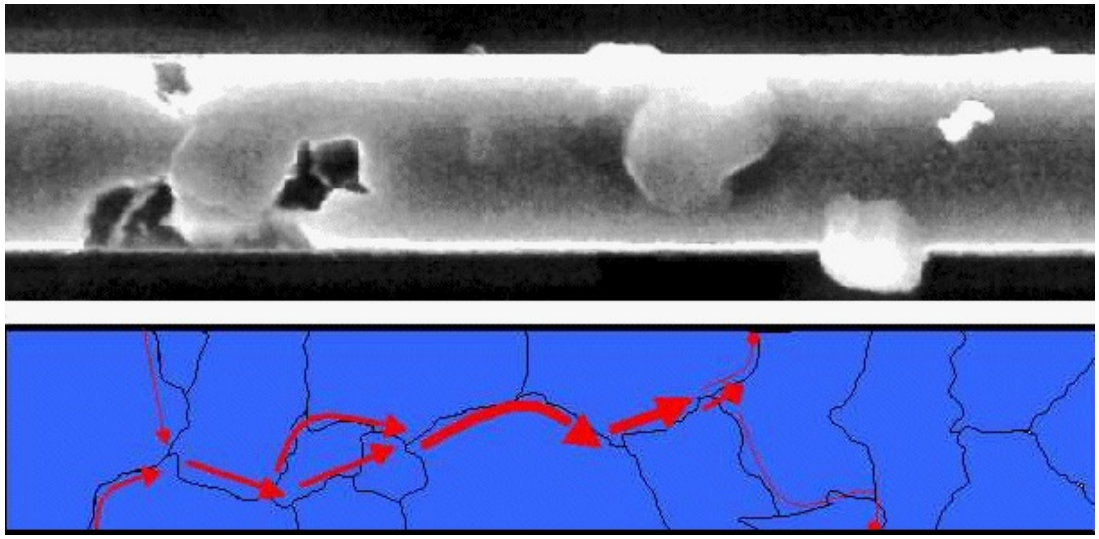



**Figure 3.25** Metal interconnect layers

# Metal interconnect layers

- Metal to metal: via
- Metal to the gate: gate contact
- Metal to drain/source: active contact

# Electromigration



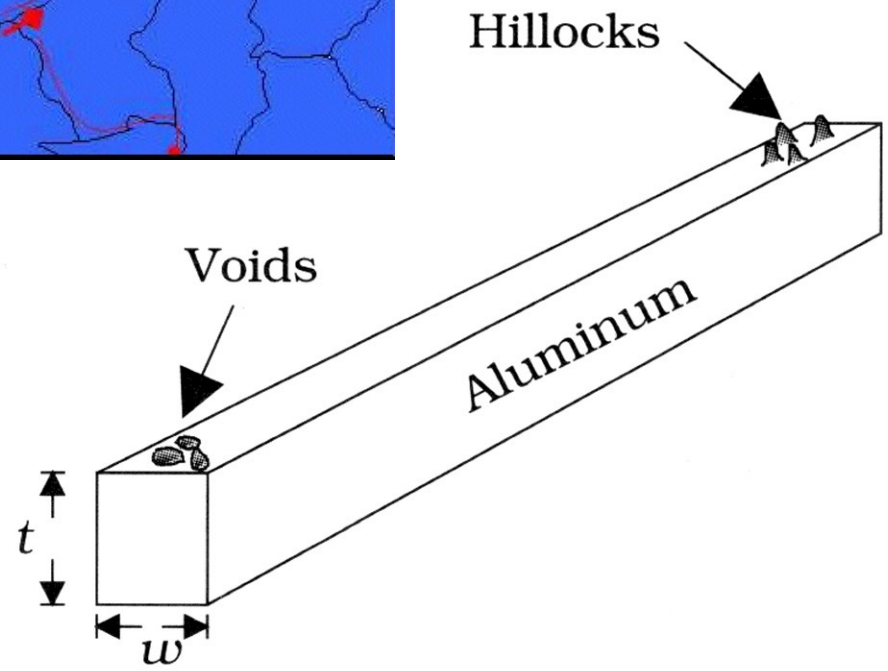



**Figure 4.4** Visualization of electromigration effects in aluminum.[1]

# Electromigration

- High current flow densities tend to literally move atoms from one end and creates pits called voids

- J= I/A where A is the cross-sectional area. Electromigration is controlled by specifying the minimum line width *w* needed to keep J below a maximum value $J_{max}$

- Copper has been introduced as a replacement to aluminum. Resistivity is one-half the value of Al.

# Questions?