Introduction

Need simple models to estimate system performance in terms of signal delay and power dissipation.

Issues include:

- Resistance, capacitance and inductance calculations.
- Delay estimations.
- Determination of conductor size for power and clock distribution.
- Power consumption.
- Charge sharing mechanisms.
- Design Margining.
- Reliability.
- Effects of scaling.

Resistance Estimation

The resistance of a uniform slab of conducting material may be expressed as

$$R = \frac{\rho}{t}\frac{l}{w} Ohms$$ where ρ = resistivity

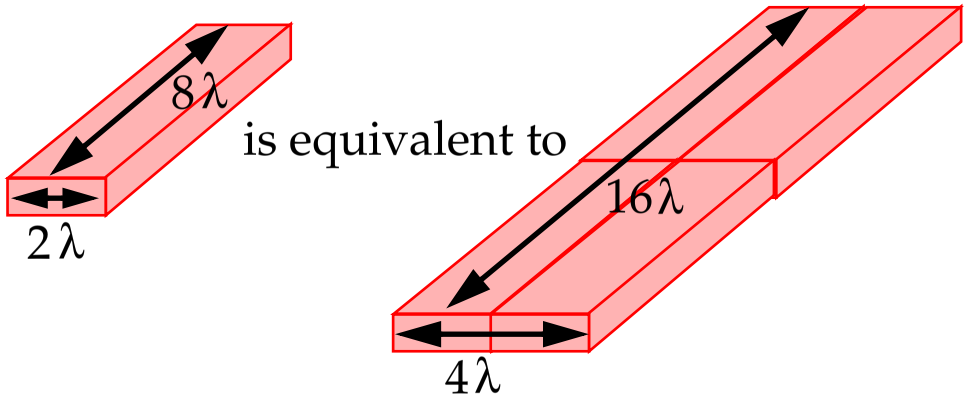
t = thickness

l/w = length/width

Alternatively as

$$R = R_S\left(\frac{l}{w}\right) Ohms$$ where R_S = sheet resistance in ohms/square.

For example, in a layout editor, such as magic or virtuoso:



Typical sheet resistances of 0.5 μ to 1.0 μ processes

material	Ω /sq
Metal1/Metal2	0.07
Metal 3	0.04
Poly	20
Diffusion	25
n-well	2K

contacts => 0.25 to 20 ohms.

Irregular shapes require more elaborate calculation - see text for examples.

Resistance Estimation

Channel resistance can be estimated in the **linear** region as:

$$R_c = \frac{1}{\mu C_{ox}(V_{gs} - V_t)}\left(\frac{L}{W}\right) Ohms = \frac{1}{\beta(V_{gs} - V_t)} Ohms$$

A range of 1,000 to 30,000 ohms/square are possible for n-channel and p-channel devices.

Typical betas for identically sized devices; n-dev: ~90, p-dev: ~30 microA/V^2.

Temperature changes both mu (mobility) and V_t (threshold voltage) and, therefore channel resistance.

Channel resistance increases with temperature, approximately +0.25% per degree C above 25 degrees.

Metal and poly resistance change about 0.3% and well diffusions about 1% per degree C.

Capacitance Estimation

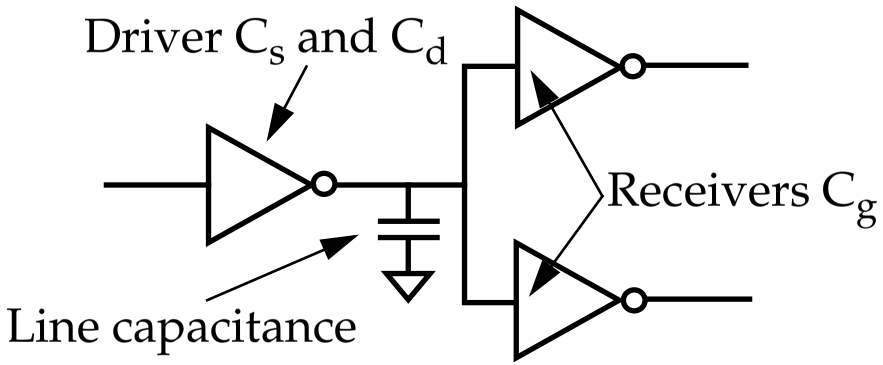
Switching speed of MOS systems **strongly** dependent:

- Parasitic capacitances associated with the MOS transistor.
- Interconnect capacitance of "wires".
- Resistance of transistors and wires.

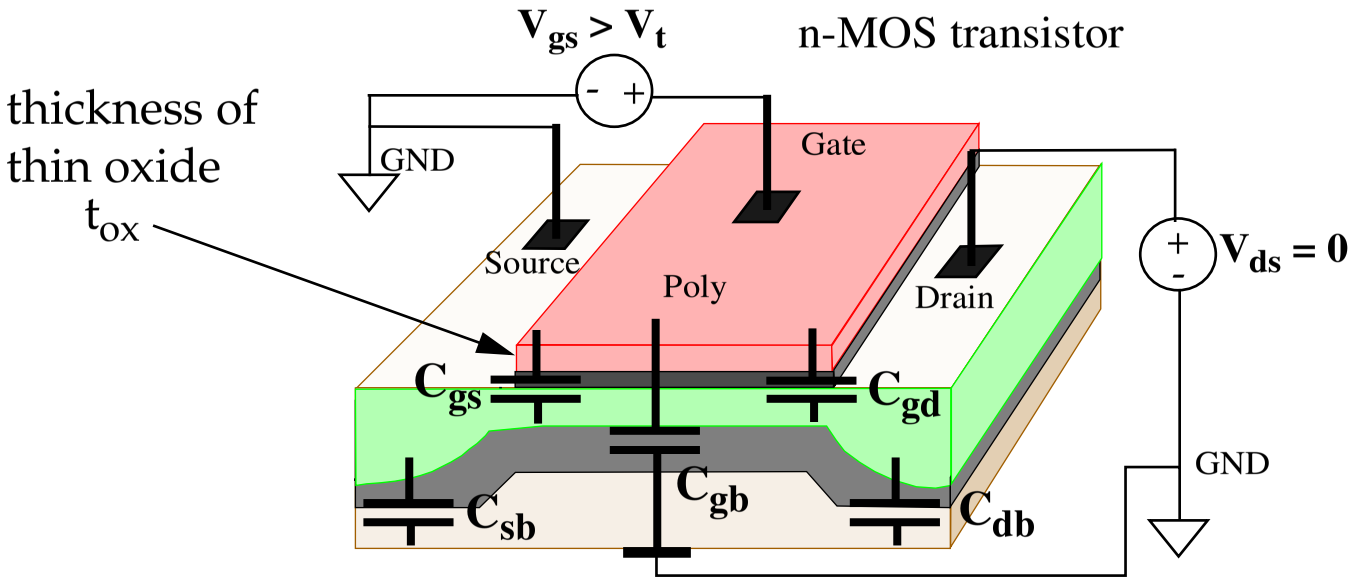
Total *load* capacitance on the output of a CMOS gate is sum of:

- Gate capacitance (of *receiver* logic gates downstream).
- *Driver* diffusion (source/drain) capacitance.
- Routing (*line*) capacitance of substrate and other wires.

Let's consider approximations of each of these capacitances and subsequent approximations of delay based on these expressions.



Estimating Gate Capacitance:



The capacitance of a MOS transistor can be modeled using 5 capacitors.

An approximation of gate capacitance (C_{gs}, C_{gd} and C_{gb}) is given as:

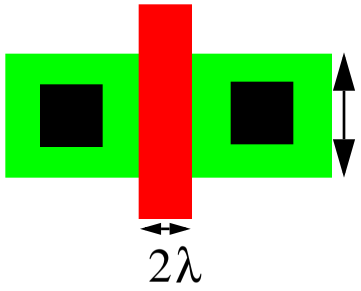
$$C_g = C_{ox}A$$

where C_{ox} is the thin-oxide capacitance per unit area, $C_{ox} = \frac{\varepsilon_{SiO_2}\varepsilon_0}{t_{ox}}$

Estimating Gate Capacitance:

For example, for thin-oxide thickness of 15 nm,

$$C_{ox} = \frac{3.9 \times 8.854 \times 10^{-14} F/cm}{15 \times 10^{-7} cm} = 2.3 fF/um^2$$



In λ = 0.5 technology, W = 2 and L = 1

$$C_{g(intrinsic)} = 2\mu m^2 \times 2.3 fF/\mu m^2 = 4.6 fF$$

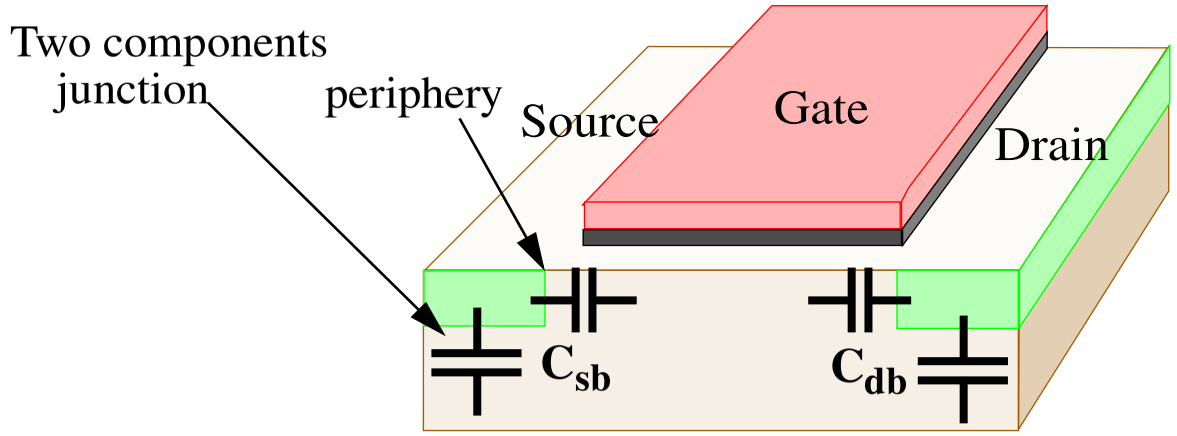
Typical value for a 1 micron process: $1800 aF/\mu m^2$

This is a conservative estimate of gate capacitance that does not include fringing fields (extrinsic) gate capacitance.

Gate capacitance increases as the thin-oxide thins.

Estimating Source/Drain Capacitance:

An approximation (**lumped model**) of source/drain capacitance (C_{sb} and C_{db}) is given as:



$$C_d = C_{ja} \times (ab) + C_{jp} \times (2a + 2b)$$

where C_{ja} = junction capacitance per μm^2

C_{jp} = periphery capacitance per μm

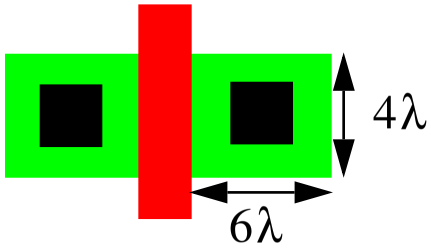
a = width of diffusion region (μm)

b = length of diffusion region (μm)

This model assumes a zero DC bias across the junction.

Estimating Source/Drain Capacitance:

For example:



n-channel device

Typical values for 0.5 micron process

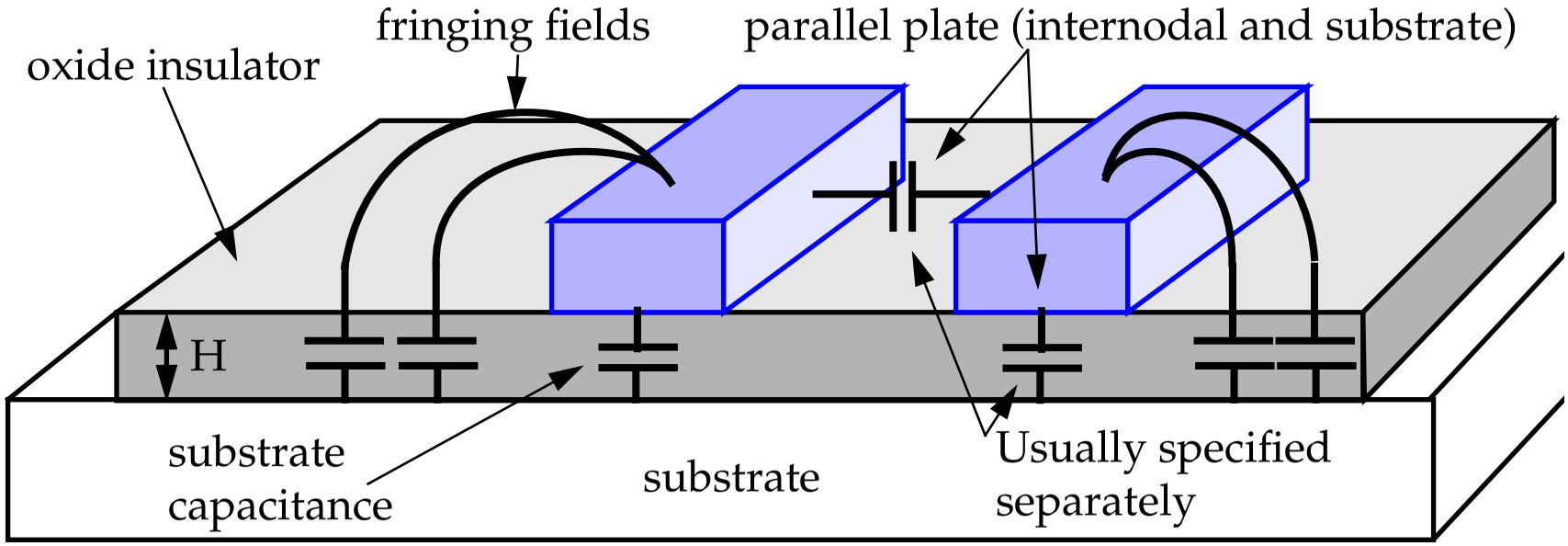
	n-device	p-device
C_{ja}	$0.04\,fF/\mu m^2$	$0.17\,fF/\mu m^2$
C_{jp}	$0.3\,fF/\mu m$	$0.2\,fF/\mu m$

$$C_d = 2 \times 3 \times 0.04\,fF/\mu m^2 + (2 \times 2 + 2 \times 3) \times 0.3\,fF/\mu m = 3.24\,fF$$

Because of fan-out, **gate** capacitance usually dominates the loading.

Estimating Routing Capacitance:

Routing capacitance between metal and poly can be approximated using a parallel-plate model.



$$C_{p-p} = \left(\frac{\varepsilon}{t}\right)A$$

where ε = permittivity of the insulator
t = insulator thickness
A = area of the parallel-pate capacitor

or

$$C_{p-p} = C_S A$$

where C_S is substrate capacitance per unit area.

The effect of the fringing fields is to increase the effective area of the plates.

Simple Gate Delay Model

Appropriate if the wire delay is MUCH less than the gate delay, e.g.,

$\tau_w \ll \tau_g$ or $l \ll \sqrt{\frac{2\tau_g}{rc}}$

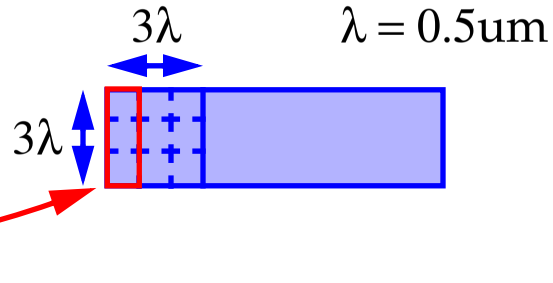
In this case, we model the "electrical node" simply as a capacitive load.

This expression derives from the expression for RC delay (we'll see this later).

As an example, assuming gate delay is 200ps, what is the maximum length of a minimal-width metal wire (in 1.0um technology) that we can use without worrying about the RC delay of the wire itself?

Assume Metal1 = 0.05 Ohms/square and 30 aF/um^2.

$$l \ll \sqrt{\frac{2 \times 0.2 \times 10^{-9}}{\frac{0.05}{3} \times \frac{30 \times 10^{-18} \times 3}{\lambda^2}}} = 16,330\lambda$$



But this assumes there is no gate load capacitance.

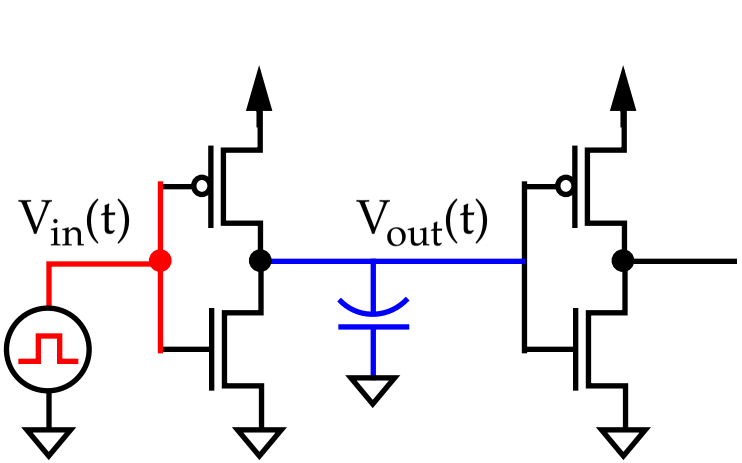
A conservative estimate is 5000 lambda (~16,330/3).

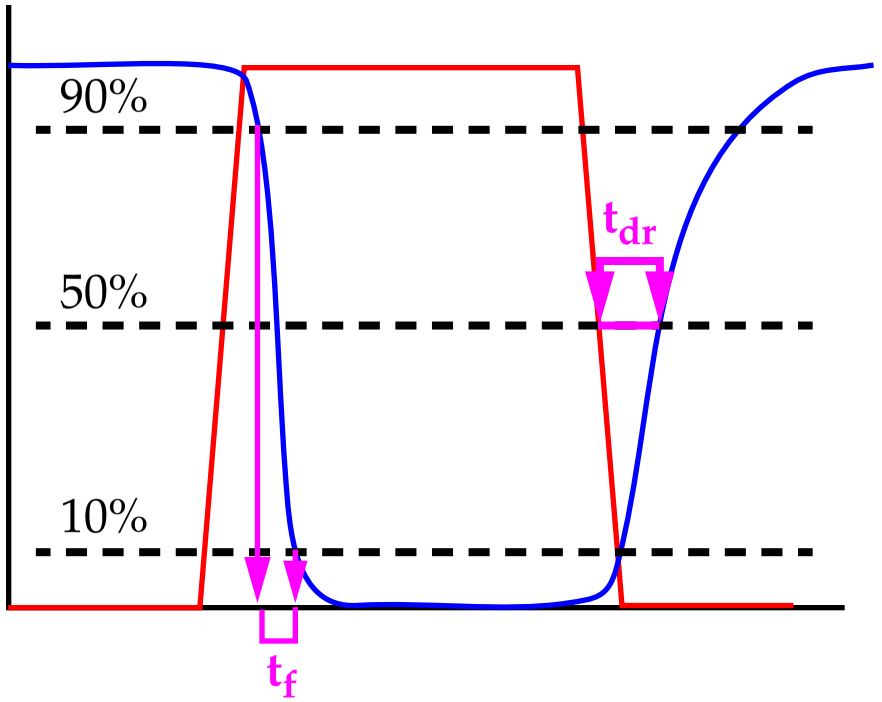
In a 1.0um process, RC delay MUST be considered for any wire > 2.5mm.

Simple Gate Delay Model

But for now, let's consider "electrical nodes" for which we can ignore distributed RC effects.

Our model and definitions:





Fall/rise time, e.g. t_f, computed between 10% and 90% of V_{DD}.

Propagation delay, t_{dr}, computed at 50% points on input and output waveforms.

Simple Gate Delay Model

How do we model gate delay?

Assume input is driven by a step waveform (unlike previous slide).

Approximation for fall time:

$$t_f = k \times \frac{C_L}{\beta_n V_{DD}}$$

where k = 3 to 4 for values of V_{DD} = 3 to 5 V
and V_{tn} = 0.5 to 1.0 V.
and C_L (load capacitance)
includes drain cap + line cap + gate caps

If $\beta_n = \beta_p$ then $t_r = t_f$ (e.g., p-trans are twice as wide as n-trans).

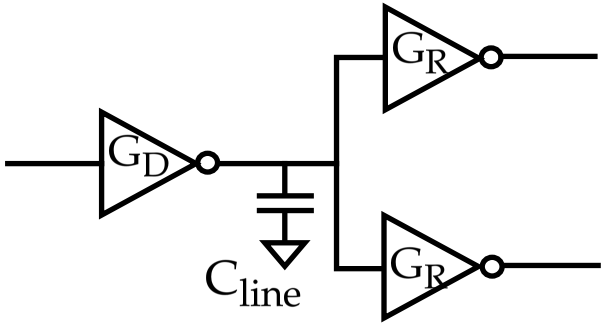
Also $t_{dr} = t_{df} = \frac{t_r}{2}$ (since delay is usually dominated by **output** rise/fall times).

And average $\tau_g \approx \frac{t_{df} + t_{dr}}{2}$

Note that the input waveform's finite slope will also effect this result -- adding a small amount of additional delay which is ignored here -- see text for details.

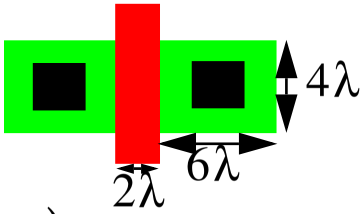
Simple Gate Delay Model

For example, let's compute the delay between G_D and G_R:



n-transistor dimensions.

p-transistor = 2*n-tran width.



Given: $C_{line} = 800aF$ (Includes Metal 1 and poly caps).

$$\beta_n = \beta_p = 90\frac{\mu A}{V^2} \text{ and } k = 3$$

Previously, we computed the drain and gate cap for an n-transistor as:

$$C_{dn} = 3.24\,fF \text{ and } C_{gn} = 4.6\,fF$$

In a similar way, we can compute drain and gate cap for a p-transistor as:

$$C_{dp} = 4.84\,fF \text{ and } C_{gp} = 9.2\,fF$$

then

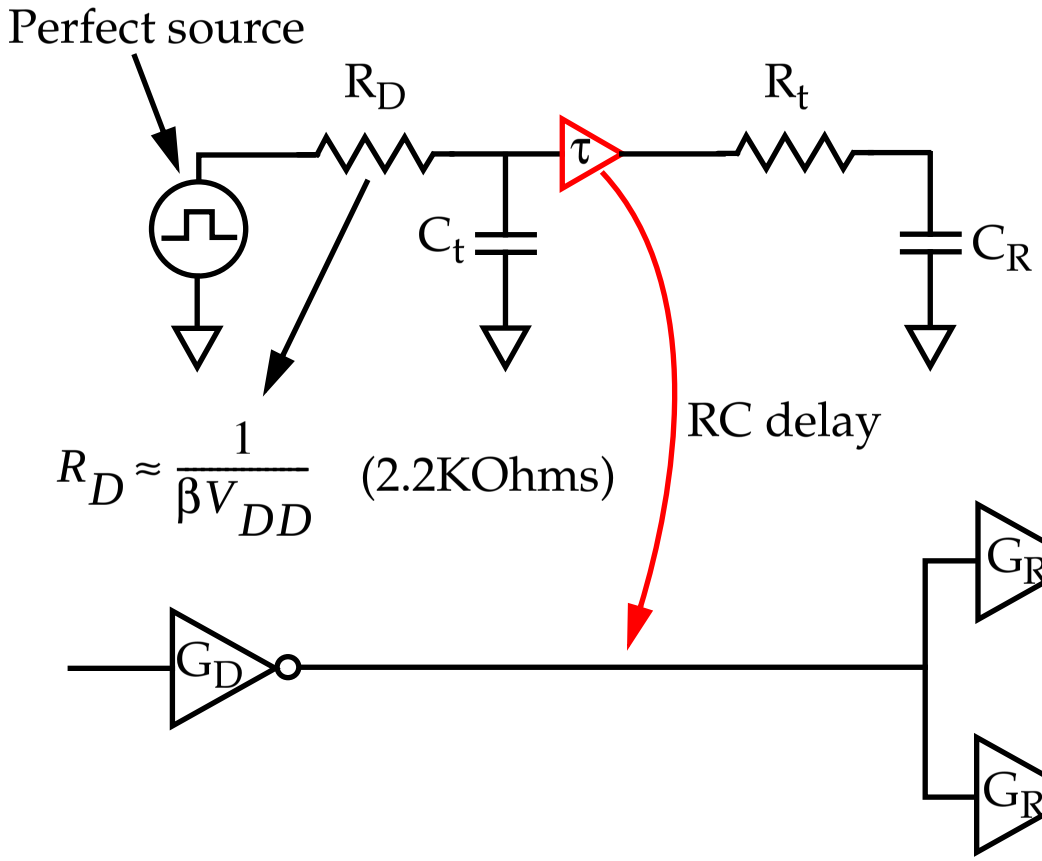
$$C_L = 3.24\,fF + 4.84\,fF + (2)(4.6)\,fF + 2(9.2)\,fF + 0.8\,fF = 36.48\,fF$$

$$\tau_g = 3 \times \frac{36.48\,fF}{90\frac{\mu A}{V^2}(5V)2} = 122\,ps$$

Distributed RC effects

If the wire delay ~= gate delay, then we will have to use a different approximation consisting of three components:

When $\tau_w \approx \tau_g$



Driver and receiver loading:

R_D = Output resistance of the driver.

C_t = Total lumped cap. of the line (no gate).

R_t = Total lumped resistance of the line.

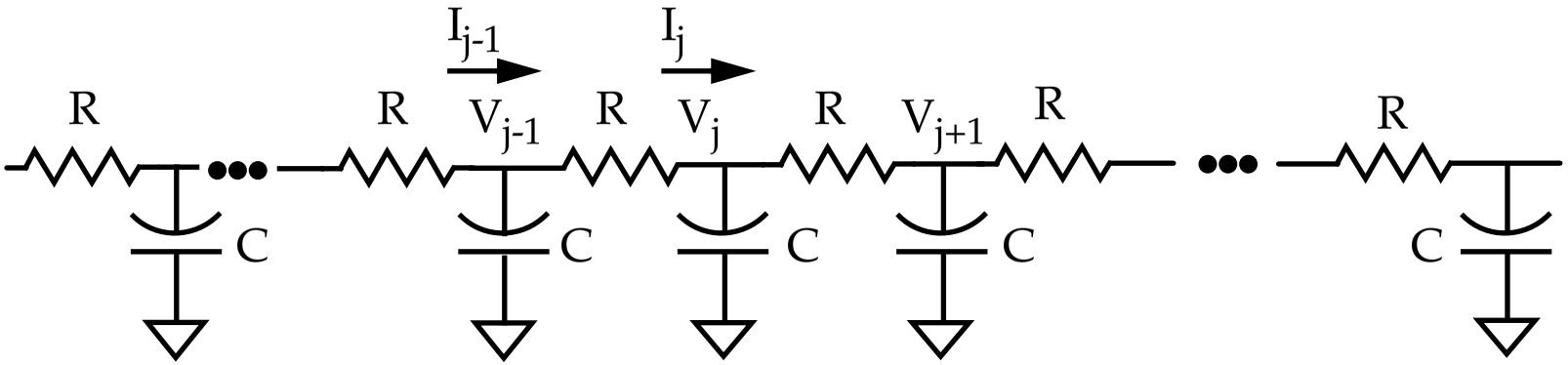
C_R = Input capacitance of the receiver (gate cap).

Yields three values, add them together to get delay.

τ is the distributed RC delay explained below.

Distributed RC effects

A wire can be represented in terms of several RC sections:



A discrete analysis of this circuit yields an approximate delay of:

$$t_n = RC\frac{n(n+1)}{2}$$ where n = number of sections

As n becomes large (and the sections becomes small), this reduces to:

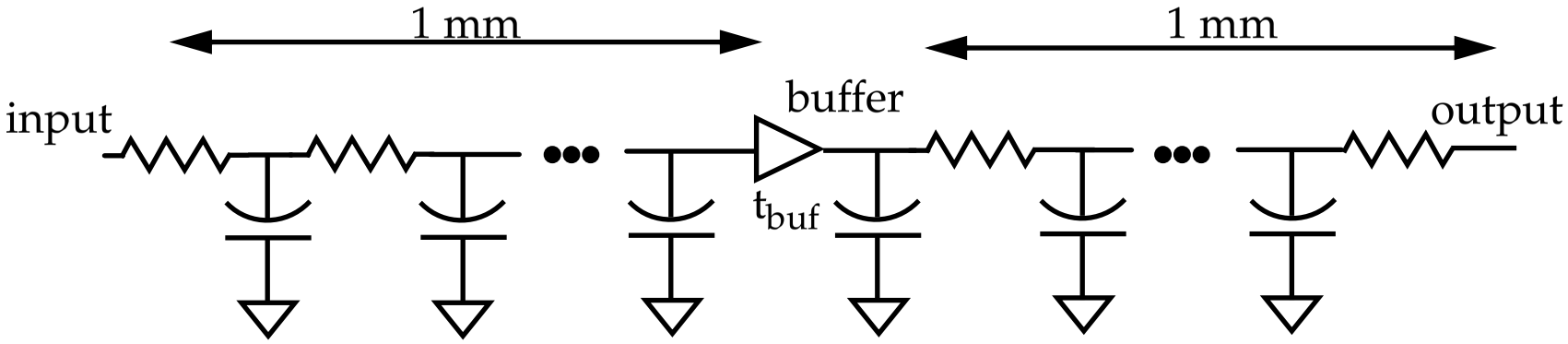
$$t_1 = \frac{rcl^2}{2}$$

r = resistance per unit length
c = capacitance per unit length
l = length of the wire

RC effect dominates for very long wires due to l^2 term, e.g., doubling the length of the wire, quadruples the delay.

Distributed RC Effects

For example, consider a long poly wire:



The buffer is one possible method of reducing the propagation delay.

Assume r = 20 Ohms/micron and c = 0.4 fF/micron, then:

With the buffer:

$$t_p = \frac{rcl^2}{2} = 4\times10^{-15}(1000)^2 + \tau_{buf} + 4\times10^{-15}(1000)^2 = 8ns + \tau_{buf}$$

Without the buffer:

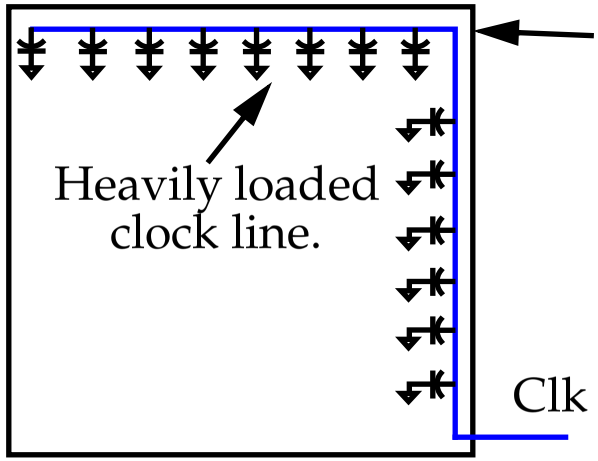
$$t_p = \frac{rcl^2}{2} = 4\times10^{-15}(2000)^2 = 16ns$$

The buffer version is faster if its delay is less then 8ns. This is easily achieved.

Distributed RC Effects

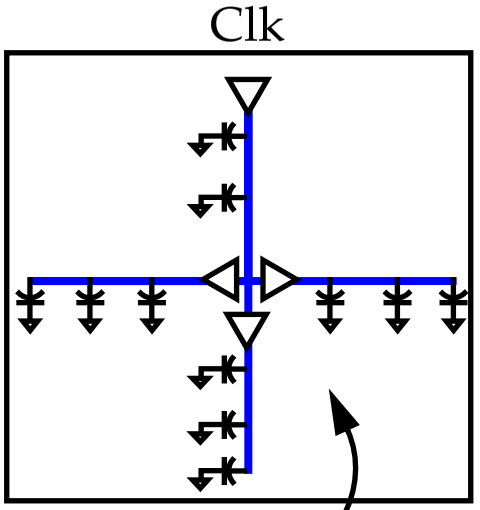
When are distributed RC effects important to consider:

- Long wires with high resistance, e.g. poly wires.
- Long, heavily loaded clock lines.



Can put buffer in to help (see previous example).

Not a good idea



Clock skew can be reduced by:

Reducing the effective length between the driver and receiver gates.

Adding buffers.

Widening metal (which increases C/unit area by a little bit (since it is already heavily loaded) but reduces R.

Distributed RC Effects

An example showing that reducing R at the expense of C helps a lot in some cases:

- Assume clock wire runs over 20mm and 50pF is distributed evenly along the line.
- Assume r = 0.05 Ohms/um.

Then clock skew (delay to the end of the wire) is:

$$t_p = \frac{rcl^2}{2} = \frac{\frac{0.05\Omega}{\mu m} \times \frac{50pF}{20,000\mu m} \times (20000)^2}{2} = 25ns$$

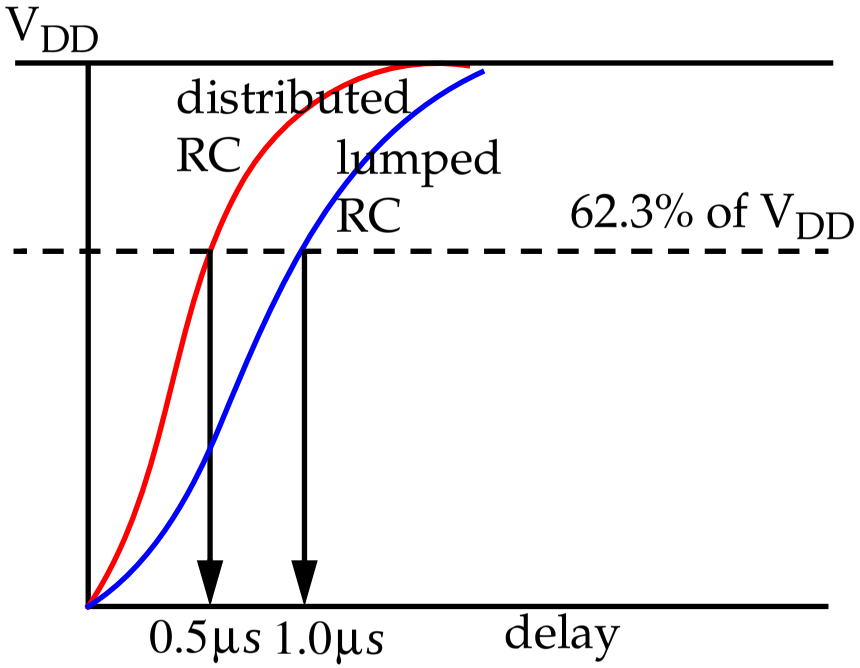
Solutions include adding a buffer, distributing the clock from the top center and/or widening the metal wire.

For example, reducing l to 10mm and widening the clock wire to 20um:

$$t_p = \frac{rcl^2}{2} = \frac{\frac{0.05\Omega}{20\mu m} \times \frac{25pF}{10,000\mu m} \times (10,000)^2}{2} = 0.31ns$$

Distributed RC Effects

How does the distributed RC model differ from lumped model?



The lumped version is conservative by a factor of 2.

Usually, conservative models are preferred, particularly in cases which are difficult to approximate accurately otherwise.

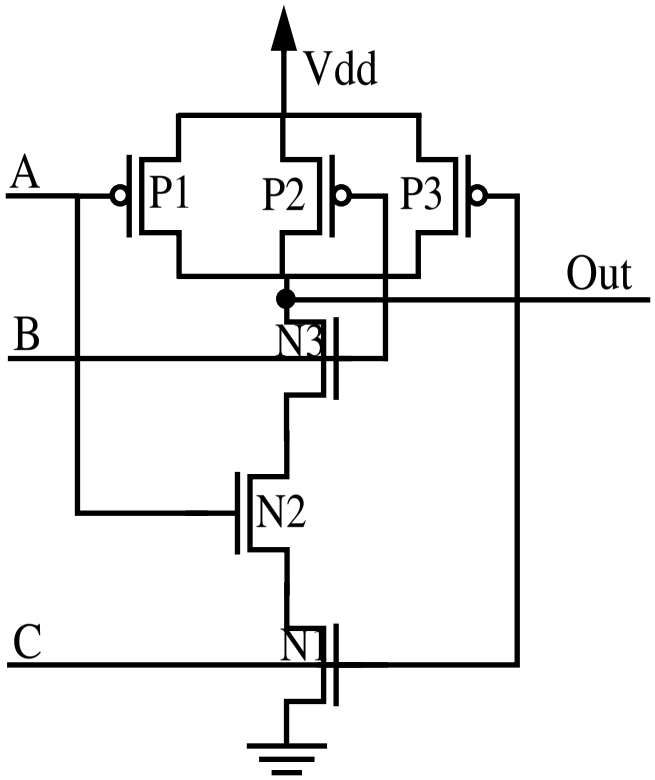
Of course, since the distributed model is simple and is more accurate in this case, it is preferred.

Note that these effects are completely ignored in the simple gate delay model.

FYI: We estimate delay using RC time constants assuming that the time taken for a signal to reach 62.3% of its final value approximates the switching point of an inverter.

Gate Delays

Construct an equivalent inverter, e.g.,



3-input NAND

Assume $W_p = W_n$

$$\beta_{neff} = \frac{1}{\frac{1}{\beta_{n1}} + \frac{1}{\beta_{n2}} + \frac{1}{\beta_{n3}}}$$

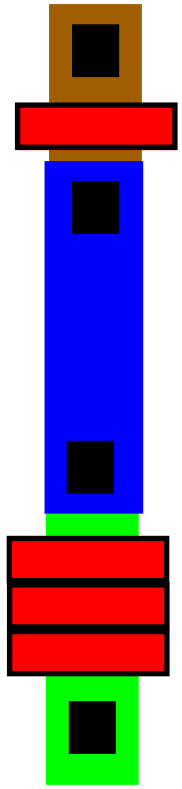
If $\beta_{n1} = \beta_{n2} = \beta_{n3}$

then $$\beta_{neff} = \frac{\beta_n}{3}$$

For the pull-up case, only one p-transistor has to turn on:

$$\beta_p = 0.3\beta_n$$

$$\frac{t_r}{t_f} \approx 1$$



3-input NANDs are closely balanced since n beta is about 3 times larger than p beta.