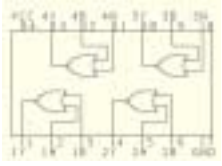
Lecture #2: Digital Logic Implementation

Paul Hartke
Phartke@stanford.edu
Stanford EE121
January 10, 2001

Review and Misc

- Last time we discussed switching algebra, boolean algebra, and (to a lesser extent) Karnaugh Maps.
- At the logic level this allows us to design various functions.
- How do we realize them in hardware?
- In E40 we used discrete components (ie, 7400 series chips) that for example each contained 4 NOR gates.

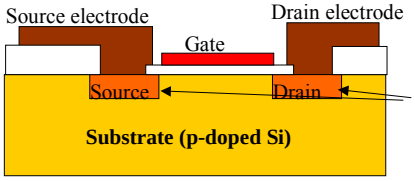


What is inside these chips?

- Transistors of course!!!!
- In lab 2 you will deal with bipolar transistor circuits (Transistor-Transistor Logic, TTL) as well as CMOS transistor circuits. TTL is more robust, usually faster, but more "expensive" and power hungry. During the last 20 years MOS circuitry has become fast enough for nearly all applications. That is all the detail we need for EE121.

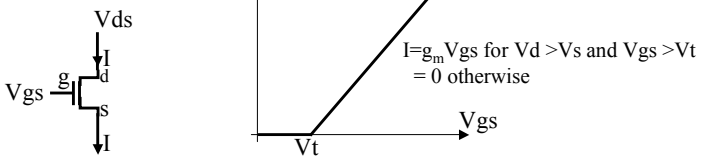
NMOS Transistor

- At its very simplest, NMOS can be treated as an electrically controlled switch. That is, the resistance between source and drain, Rsd = 0 when Vg-Vs > Vt (threshold voltage) and = ∞ otherwise ('zero-order model').



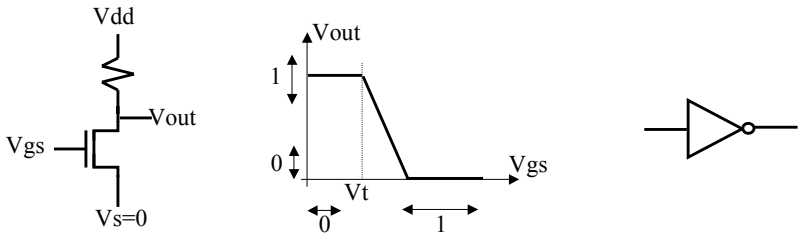
NMOS as Voltage Controlled Current Source

- Two normal operating regimes of transistor:
  - Amplifier (Analog)
  - Switch (Digital)
- Surprise, Surprise we only use digital mode where the transistor is either "on" (conducting) or "off" (not conducting)



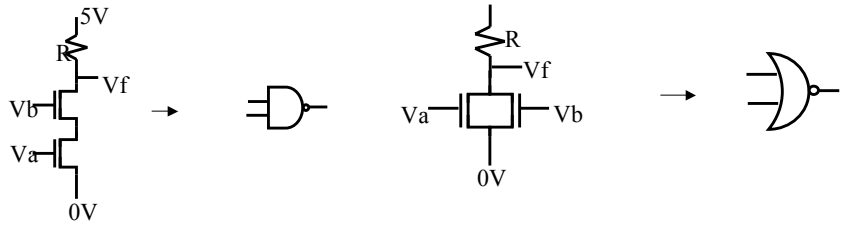
How does this become useful logic?

- Start small—the Inverter
  - The ranges of '0' and '1' at the output and at the input must obviously be the same. We must leave a region between the two ranges to allow for variations in Vt and other transistor parameters.
    - Noise Margin



Inverters not so interesting…

- We can extend the inverter to NOR and NAND gates as shown below.

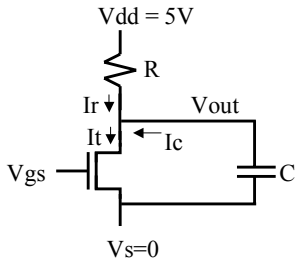


Power to the People!?!

- What is power consumption when Inverter is "pulling down" (1 at input, 0 at output)?
  - Since the transistor is on, there is a path from power to ground and so device has static power consumption.
  - How much? Depends on the intrinsic "on resistance" (a device characteristic) of the NMOS.

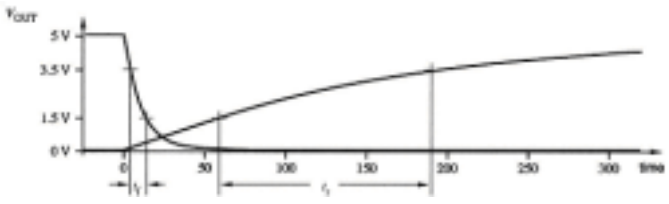
Switching Speed?

- Only useful to talk about speed when "loading" is defined.
  - Good figure of merit is "Fan out of 4" inverter (FO4) speed.
- Assume capacitive load for the moment
- Basically RC circuit
- How do those look?



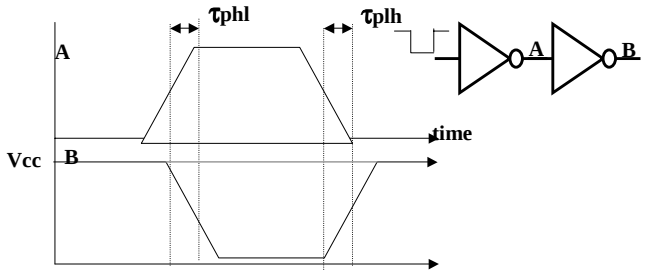
Propagation Delay Approximation

- This curve is non-trivial to calculate exactly so we won't (benefits of engineering approximation! ☺)
- If sharp enough then approximate as a delay (good approximation for most things).
- Does matter for chip I/Os though…



Propagation Delay (tp) in Action

- Tp a property of the devices and given in the process (we'll look at specific values used in this class a little later).

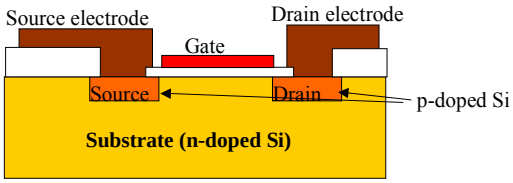


Eliminating Static Power…

- The static power comes from the pull-up resistor. What if we could replace it with a transistor that was the inverse ("complement") of the NMOS?
  - Still assume capacitive load…
- This would mean only one device is conducting at a time—no more static power but only dynamic power
  - Call it "Complementary MOS" →CMOS

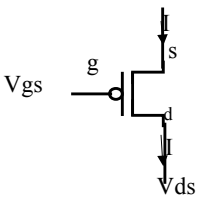
PMOS Transistor

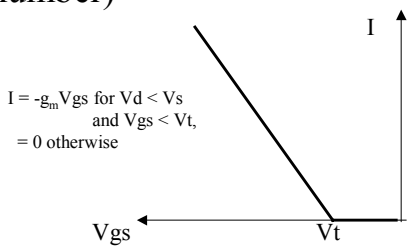
- Operates as the inverse of the NMOS transistor (except that you need one twice as big to get the same operating characteristics).



PMOS Transistor 2

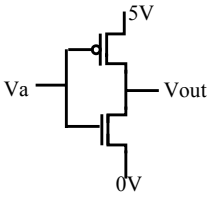
- All the characteristics of MOS hold
- Vgs < Vt
  - (Vt is a negative number)

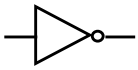




CMOS Inverter

- No static power consumption unless resistive loads are added (including LED's).
- All outputs drive to the power supply 'rails' (e.g. 0 and 5V) – devices are cascadable
- If the transistors are sized properly the switching threshold will be exactly halfway between Vdd and gnd. (5 and 0V).



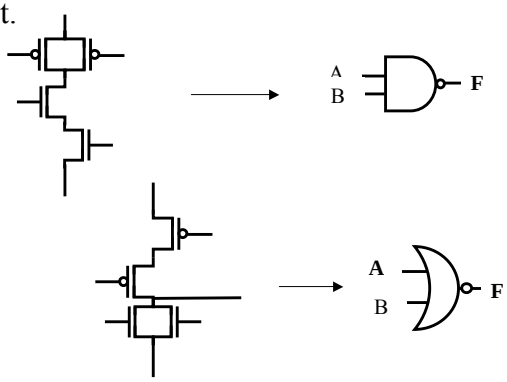


Gates a plenty

- Can make all sorts of gates from this according to the following rules
- In the steady state, either Vdd or Gnd (but ***not*** both) must connect to the output through 'on' PMOSTs or NMOSTs respectively.
- All inputs to a circuit and all intermediate outputs feed only into the gates of other transistors.
  - Thus providing only capacitive loads

Two Common Gates

- On the diagrams on the left, show the inputs, outputs and power supply connections to realize the functions on the right.



Non-Inverting Outputs?

- How do we make non-inverting outputs for a basic gate in CMOS?

Power Calculation

- For each transistor how much power is being consumed?
- If it is only driving a capacitive load, then we can approximate that all the power is consumed in the act of switching. It is nontrivial to calculate the power via the resistance of the on transistor so use the output capacitance.
- Power to charge/discharge a cap: $\frac{1}{2}CV^2$
- Assume circuit is driven by a clock (one up and one down transition) so it is CV^2 each clock cycle
- If the clock frequency is f, then equation becomes:

$$\underline{\boldsymbol{CV^2f}}$$

Power Comments

- As you go faster (f up), you use more power.
- This assumes each gate transitions on each clock edge (which we know is not true).
  - Switching factor needed for more accuracy
- C is based on
  - Size of the design
  - Process technology you are in.
- Lowering V gives a big win but if you lower too low, second order effects start becoming prominent.
- We will do NOTHING in this class to address power consumption (except not design unused logic)…

We won't be doing CMOS design in EE121

- Designing with the actual transistors ("full custom design") is very complex and expensive so we won't be doing it (take EE271—it's very good).
- Using discrete components as in E40 leaves a lot of wiring drudgery so we don't want to do that either.
- Field Programmable Gates Arrays (FPGAs) to the rescue!!!!
  - They are built from CMOS logic which is why we covered today's material.