Physical Model of NMOS

Current conduction through electron “channel” between source and drain

Cross Section of a NMOS Transistor

Apply Positive Gate Voltage

Forming a S-D Channel Due to “Charging” the Gate Capacitance

Circuit Model of a NMOS Device

Metal
Why Do We Need Semiconductor?

• Can we use the induced charge in a metal capacitor for transistor?
Semiconductor – Silicon

• Intrinsic (pure Si): like an insulator
• Can be made conducting by “doping”
  – N-doping: introduce group-V elements like As or P
    • Extra electron to conduct current
  – P-doping: introduce group-III elements like Ga
Resistivity Range of Doped Si

- Range of resistance:
  
  \[ p: 10^{-3} \sim 10^3 \, \Omega \cdot \text{cm} \]

- In comparison:
  - Metal has much lower resistivity
    - Aluminum: \( 2.6 \times 10^{-6} \, \Omega \, \text{cm} \)
    - Copper: \( 1.7 \times 10^{-6} \, \Omega \, \text{cm} \)
  - Insulator has much higher resistivity
    - \( \text{SiO}_2: \sim 10^{17} \, \Omega \, \text{cm} \)
Applications
ADCs and DACs

Voice

Motor

Analog-to-Digital converter

Digital Signal Processing

...110011100...

...100110100...

Driver

Wireless

ADC → Amplifier

DAC → Speaker
R-2R Ladder Digital-to-Analog Converter (DAC)

How to set all these “digital” voltages?
Remember **Superposition** and **Equivalence**?

Use superposition: Start with first voltage source:
R-2R Ladder Digital-to-Analog Converter

Adding all contributions from the sources

\[ V_{\text{out}} = \frac{V_{b_0}}{2^N} + \frac{V_{b_1}}{2^{N-1}} + \cdots + \frac{V_{b_N}}{2} \]

\[ V_{\text{out}} = \frac{1}{2} V_{b_N} \]

Adding all contributions from the sources
Switches

RC time of 2 gates

\[ R_C \text{ time of 2 gates} = R_{ON} (C_p + C_n) = \tau \]

\[ \uparrow \quad R_p, R_n \]
How fast can we “convert”?

- If there were no capacitors, we could do it instantly!
Analog to Digital Conversion

- Very fast massively parallel architecture
- Requires $2^N$ comparators (specialized op-amps)
- Op-amps have input capacitance
- Power consumption is high for fast operation

This creates a "Thermometer" digital code. Need to convert to binary for most applications.
As another example, let's say that we are 5.8 hours behind in lecture (which we will represent as 5.8V), and we would like to convert that number into its digital representation so that we can store it easily. We have access to a 3-bit SAR ADC with a reference voltage $V_{\text{ref}}$ of 8V. Figure 6 shows the output of the DAC in the ADC as the algorithm progresses and tries to find the closest match to the 5.8V input we feed it.

In the first step, you can see the MSB turns on and produces a DAC voltage of 4V, which is half of $V_{\text{ref}}$, as expected. The comparator tells us that this voltage is still less than the input voltage of 5.8V, so we leave this bit on. In the next step, we turn on the middle bit, which produces a step of 2V ($V_{\text{ref}}/4$). But now, the combined step of the MSB and middle step produces 6V, which is greater than our 5.8V input. The algorithm detects that we've overstepped and turns the middle bit off before proceeding onto the final bit, the LSB. In the final step, the LSB turns on, producing a step of 1V ($V_{\text{ref}}/8$), and the combined step of the MSB and LSB produces a 5V DAC output. The comparator once again tells us that this is less than the input voltage, but we're out of bits we can work with, so the algorithm terminates. The closest 3-bit digital representation we can get for this 5.8V is 101, which corresponds to an analog voltage of 5V. If we had more bits, we would be able to more closely approximate the true input voltage and get a better representation.

Below is a figure of how we will implement this ADC in the lab using our Arduino. Our ADC will have 4 bits of resolution, as shown in the image below. The Arduino handles the SAR logic (executed in code) and the 4 pins each represent one bit, each taking on a voltage of either 0V or $V_{\text{ref}}$ (which is 5V for the Arduino).

![ADC Diagram](image)

- Use a DAC to guess signal and find best digital representation
- Can do this in $\log(N)$ steps ("guessing game")