

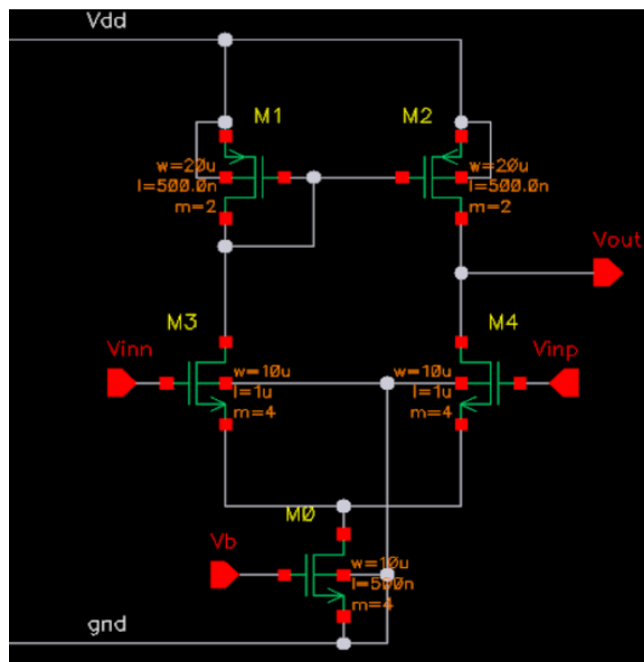
Project-3 (20%) – Common Centroid method (CCM)

In analog layout, device matching is necessary and critical. Before we use common centroid method to place the transistors as granted, can we create different layout patterns to show indeed CCM is better? (Note: please read an introduction of differential pair to understand its basic operations.

<https://www.allaboutcircuits.com/technical-articles/the-basic-mosfet-differential-pair/>)

We are going to use 0.18um design kit for this project. Please follow a design below to create two layouts. As we can see, there are 5 devices: M0, M1, M2, M3, M4. In order to let the differential pair (M3, M4) function well, M3 and M4 need to be matched. M1 and M2 also need to be matched.

Before doing layout, let's simulate this design. Note: the device sizes shown in the picture were for some tech node. They may not be good for 0.18um. Please adjust the transistor sizes (i.e., we need to do pre-layout simulations to find the proper transistor sizes) so that **Vout can perform well**. Let's label this result (A).



To get a spice netlist for simulation, either we can use a tool, such as Virtuoso to do that or simply use editor to create a spice netlist for the above circuit. (note: I believed Allen Chen in Prof. Marvin Chang's group was using Virtuoso.)

Now with the pre-layout simulation done, we can start the layout design. First, for M3, M4 we know they need to be matched. Please create two layouts: one following common centroid method (CCM), such as using this pattern (ABBA | BAAB – means two rows); the other just use arbitrary pattern, such as (AABB, or BBAA or (AABB | AABB)).

(Note: basically one layout pattern uses CCM; the other does not.)

Then, after connecting the devices,

- let's do DRC checking and fixing to clean the layout;

- then do LVS, RC extraction;
- then, simulate the resulting SPICE circuits.
 - o One result is from post-layout simulation on CCM layout – label it as (B);
 - o The other result is from arbitrary pattern – label it as (C).

Then we compare (A), (B), (C) to see the differences.

If the tools are doing things right, we expect (B)'s result is better than (C), since we were told CCM can produce better results. Just encourage us to question the conventional wisdom.

But if (B)'s result is about same as (C), or worse than (C), can we explain?

Then, for EDA tool development, we are tasked to think about how to help designers to do this layout. It leads to symbolic editor, which is explained in the class.

(Note: this Minnesota/Intel paper explored various patterns for CCM:

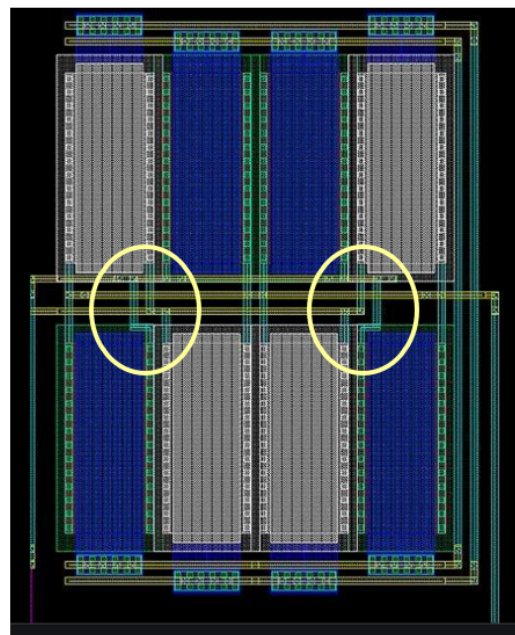
<http://www.ece.umn.edu/users/sachin/conf/date21aks.pdf>)

This project involves using tools to observe differential pair's gains based on different layout patterns.

(A purpose is to let you think about how EDA tool development can help designers to produce good layouts in this process. Basically an EDA R&D is better to understand how the tools are being used.)

In terms of **routing patterns**, welcome to refer to this picture (in a slide) and study how **poly gates** are connected, how **drain/source** are connected.

Another Device Matching Example



symmetric
routing