Lab #2

4X4 Unsigned Array Multiplier

Objective

Use SUE to design and optimize a 4x4 unsigned array multiplier and convert the design into MAX layout by using MAX Layout Generator. Then simulate the layout by using HSpice.

Design guidelines

- Refer to **Wolf 6.6** for the structure of an unsigned array multiplier;
- Always verify the logic functions of your design before doing anything else;
- Measure and minimize the critical path delay in *SUE* before doing *MAX* layout;
- Use only poly and metal 1 for routing within the multiplier;
- Use metal 2 for power and ground rails; width for each rail should be 1.5um wide;
- If you need to connect the array multiplier to other devices, use metal 1 for vertical interconnect and metal 2 for horizontal interconnect. Do not use poly;
- Your grade will be given mainly based on the size of the multiplier divided by its performance, i.e. (area)*(propagation delay), smaller the number better the grade.

Using SUE Design Manager

Use the following command to bring up *SUE*:

```
sue
```

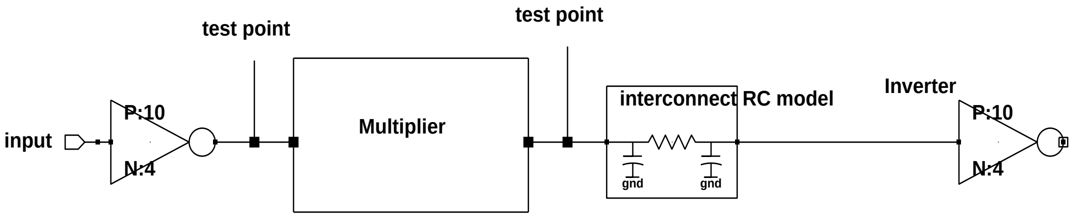
Go to Help->SUE Tutorial and run through the tutorial for information on how to use *SUE*. Pay attention on simulation part: you can verify your design logic by using one of the *SUE* simulators. Have your design logic function verified before doing anything else will save you a lot of time. There are different logic simulation programs you can use in *SUE* and you are welcome to use your favorite one.

Project procedures

1. Design an one-bit full adder icon in *SUE* and verify its logic functions;
2. Setup test environment, including building models for drivers and loads (refer to the following section for more information);
3. Construct the multiplier in *SUE* with the adder icon you built and test its logic functions;
4. Identify the critical path and use *SUE* spice simulator to measure the multiplier's propagation delay;
5. Do all you can but changing the structure of the multiplier to minimize the propagation delay. Some of the ideas may include changing the size of the adder cell, using more than one kind of adder cells, re-organizing the order of cells, etc.;
6. Once you are satisfied with your design, use *MAX Layout Generator* to convert the adders into *MAX* layout;
7. Build the multiplier with your cells; use information you collected from *SUE* as a guideline;
8. Extract the design, test and simulate it with HSpice;
9. Collect data.

Modeling driver and load

In reality, your multiplier will be part of some chip designs. Therefore, while simulating the design, you have to consider the quality of the signal that goes in the multiplier and the load circuits the multiplier connects to. As you may have already known, in high speed circuit, a digital signal is no longer a perfect square wave: it has rise and fall time delays. An easy way to generate this kind of signals is to use the output signal of an inverter (why). However, it is not so simple to model the multiplier load. This is because we have to take interconnect wire resistance and capacitance into account. Different shape of interconnect wire has different resistance and capacitance values, which greatly affect the performance of the multiplier. Please refer to **Reader pg. 115** for more information on interconnect wire modeling. The following figure is the driver and load models you should use on your design. It also indicates the location that you should acquire sample data from.



Notice that there is an inverter on the end of the interconnect wire. It is to simulate the load device capacitence that the wire is connecting to. Assuming interconnect wire is 0.9um wide, 100um long, refer to *tsmc025.pdf* for interconnect wire R/C values to use. You can download *tsmc015.pdf* from class web page http://bears.ece.ucsb.edu/class/ece124a/tsmc025.pdf. For the driver and load inverters, use P/N ratio = 10/4.

Data collecting

- Worst case delay for 4x4 multiplier;
- Critical path delay for 4x4 multiplier;
- Area of the design layout;

Design work to turn in

- Schematic for one-bit adder cell and the multiplier with highlighted critical path;
- The setup of your test environment (schematic and layout);
- Layout plot for one-bit adder cell and multiplier;
- Your own test codes (hspice, verilog, etc.);
- Printout of the waveforms with measurements;
- Proof of functionality of your multiplier (schematic level and layout level);
- Technical report to analyze the results obtained.
- Result of (propagation delay)*(layout area), unit in $(ns \cdot \mu m^2)$

Due day

In two weeks (November 4/6).