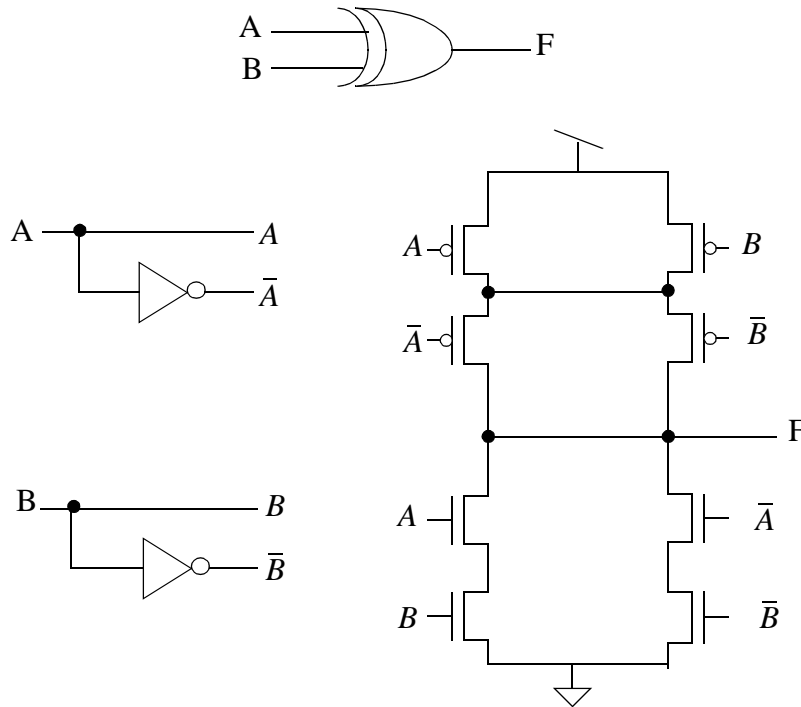


## Lab 3: XOR Gate and Programmable Ring Oscillator

Your task is to design an XOR gate and then add this gate to your ring oscillator to make it programmable. Your gate should have two inputs, A and B, and one output, F. The purpose of this lab is to improve your layout skills and introduce you to cell hierarchy in layout.



### Create a schematic

- Put the schematic above into the schematic editor
- Include instances of your inverter to create the compliments of the inputs
  - Later you will need to change the layout of your inverter. However, if you change the layout of the same cell that you are using in the ring oscillator, it will change the ring oscillator.
  - Therefore, you need to copy your inverter to a new cell name.
  - In your Library Manager select only the inverter Cell, make sure no cell views are selected
  - Go to Edit->Copy
    - In the To section type a new cell name such as *inverterXOR*
    - In the Options section make sure that Copy All Views is selected
    - Click on OK at the top
  - instance your new inverter cell into the XOR gate schematic
- In order to have the XOR the same speed as the inverters, make the transistor sizes of the FETs in the XOR twice as wide as your inverter FETs. This will make more sense later.
- Don't forget your ports.

### Create a symbol

- Your symbol should have only two inputs and one output.

## Simulate your schematic

- Create an XORtest file.
- Add your sources and perform a transient analysis to show that your circuit works.
- If you're clever, you can test the whole truth table with a single simulation.

## Layout your XOR gate

- Do your layout using the horizontal orientation for your transistors. Be sure to make the transistors the same size as was used in the schematic.
- The previous layout that you did for the inverter was a vertical orientation. Change your inverter layout so that your transistors are oriented horizontally, and match the pitch of your inverters to the XOR gate
- I would like to check your layout to give you pointers about how to make it more compact
- DRC your layout
- Extract your layout
- LVS your layout with your schematic
- When you have completed DRC and LVS successfully I will check off both of them at the same time

## Simulate your layout

- You will have to create a different stimulus file. You can start with the previous one, but you must add an additional input source.
- Don't forget to do the three step process to include the stimulus file
  - In the analog simulation window go to Setup->Environment and choose Spectre at the bottom of the form (you only have to do this once)
  - Go to Simulation->Netlist->Create Raw (you have to do this every time you change your stimulus file)
  - Go to Simulation->Netlist->Create Final

## Incorporate your XOR gate into your ring oscillator

- Create a ring oscillator schematic cell
  - hint: you do not have to instantiate all of your inverters individually
  - replace four of your inverters with your XOR gate
  - it does help in debugging to make your schematic look similar to your layout
- Simulate your ring oscillator schematic to test the control input
- In your ring oscillator layout, replace four inverters in the top row of the array with your XOR gate layout
- You should not need any additional wires to connect your XOR gate to the rest of your inverter array
- Make sure the inputs of your XOR gate are accessible to the array
- Match the pitch of the remaining inverters in that row to the pitch of the XOR layout without changing the remaining rows of your array
  - hint: Edit in Place **won't work** for this step
- Perform a DRC and an LVS check for your ring oscillator
- You **don't** have to simulate the layout

## Informal Lab Report

Turn in the following sheet with the check points signed off and the questions answered.

Name: \_\_\_\_\_

Lab Partner: \_\_\_\_\_

XOR Layout check \_\_\_\_\_

XOR DRC and LVS check \_\_\_\_\_

XOR Simulation check \_\_\_\_\_

Ring Oscillator Schematic Simulation check \_\_\_\_\_

Ring Oscillator Layout check \_\_\_\_\_

Ring Oscillator DRC and LVS check \_\_\_\_\_

## Questions and Calculations for Informal Lab Report

You may use this sheet or attach additional pieces of paper to answer the questions.

1. Sometimes in pitch matching, you end up with empty or wasted space in a cell. Why is this OK when we said it was bad to waste space?
2. One result of using standard cells is that, if you make a change to a cell, those changes are automatically done to every instantiation of this cell in your system. Describe one advantage and one disadvantage of this effect.
3. For what reason, other than pitch matching of the layout, would you have multiple copies of a cell, all of which perform the same logic function? For example, in a standard cell library you will see several different inverter cells. They are all inverters, but there is something different about each version. Hint: it has something to do with how you designed the XOR gate.