

Layout and LVS

Open-Source Analog Design Flow



UNIVERSIDAD TÉCNICA
FEDERICO SANTA MARÍA



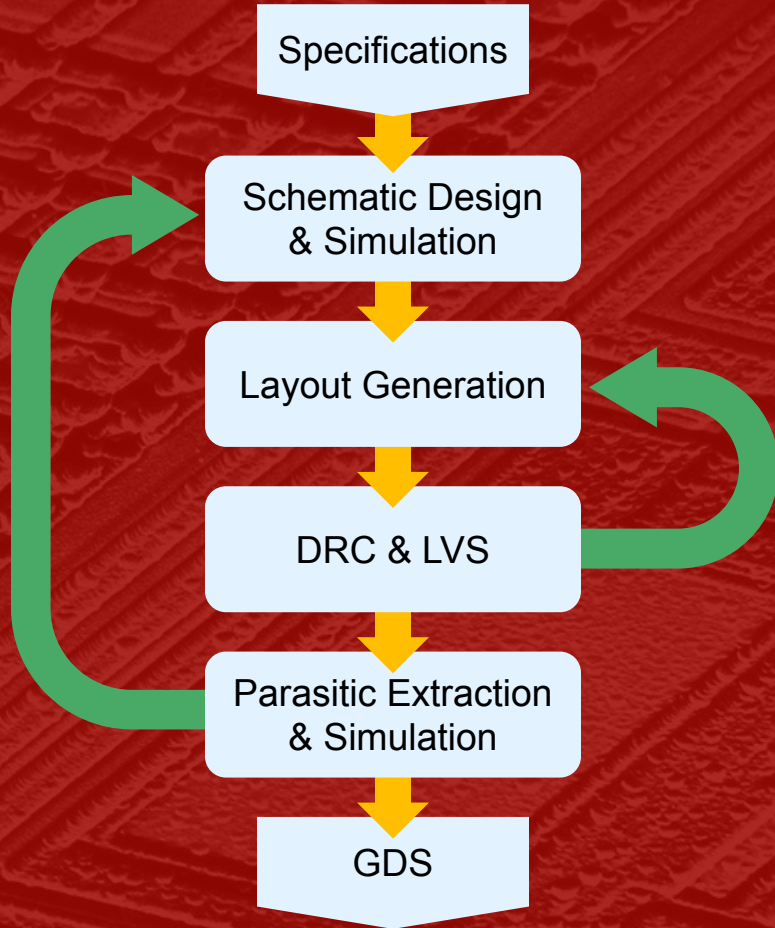
Universidad Técnica
Federico Santa María
IEEE Student Branch



CAS
IEEE CIRCUITS AND SYSTEMS SOCIETY

A Student Chapter of the IEEE Circuits And Systems Society

Design Flow for Analog Integrated Circuits

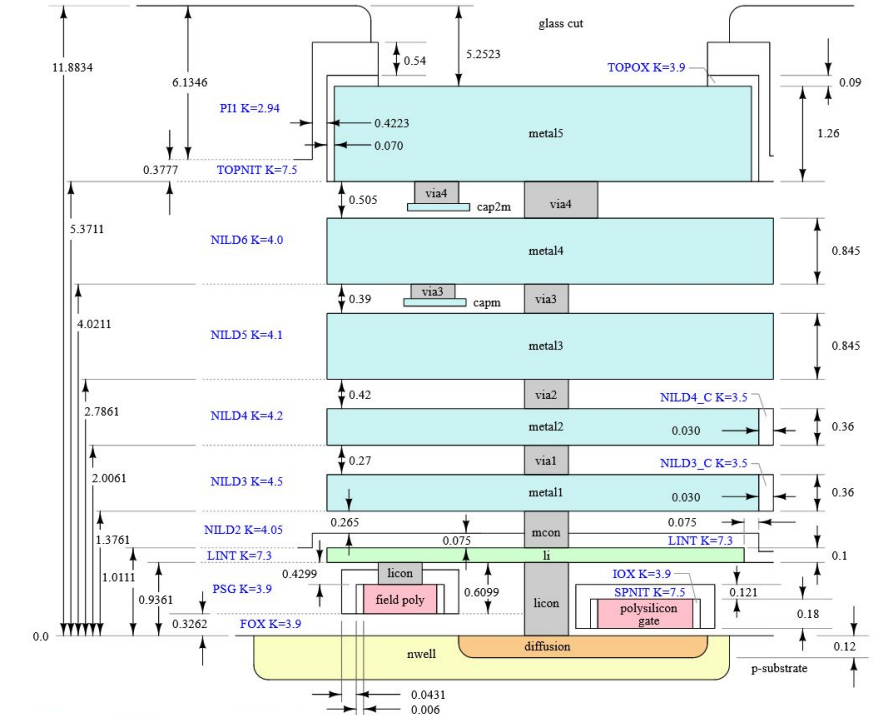


Fabrication and PDK

Process Design Kit:

Set of specifications of the fabrication process, defined by the foundry. It includes:

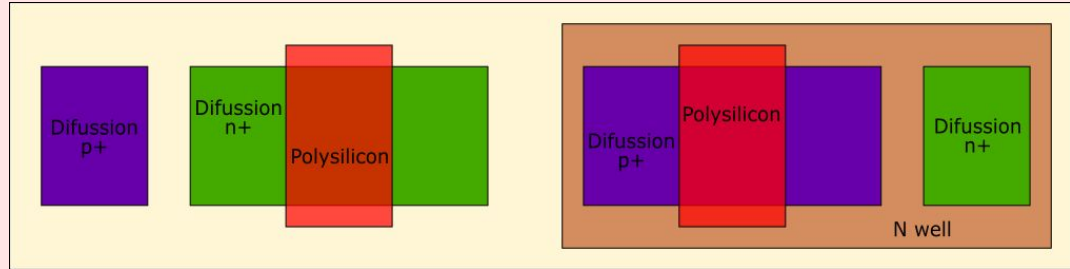
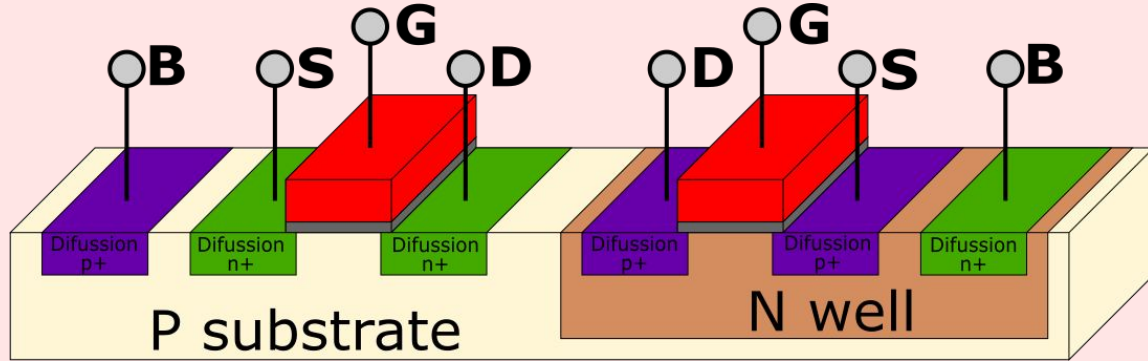
- Available layers
- Physical and electrical properties
- Device primitives
- Simulation models
- + Set of rules that ensure manufacturability of the design (Design Rule Check o DRC)



Layout

Since the vertical order of the layers is known, it is sufficient to represent the design seen from above, like a blueprint.

The shape of the polygons drawn in the different layers provide information of the physical geometry and distribution of the designed devices.



An aerial photograph of a city grid, showing a dense pattern of streets and buildings. The image is dark and has a blueish tint. The word "Layout" is written in white, bold, sans-serif font in the center of the image.

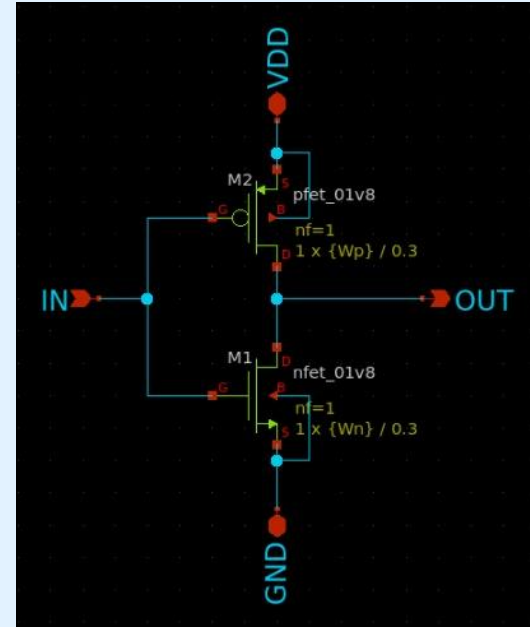
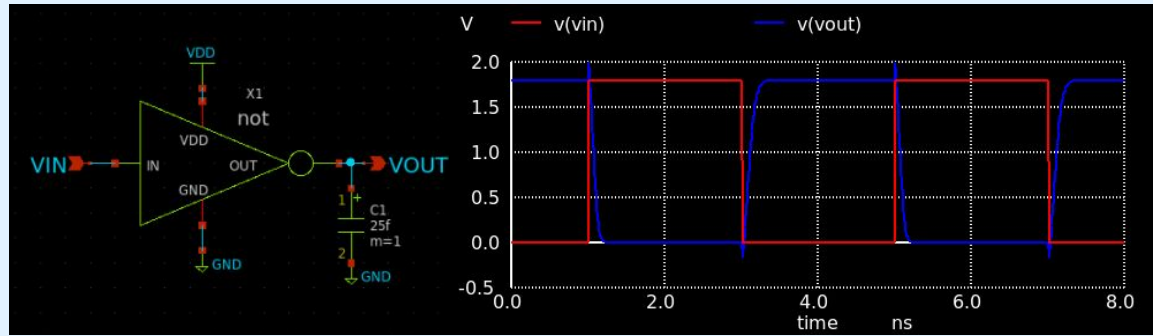
Layout

Before layout

During schematic design, once the simulations show the desired behaviour, the (preliminary) size of the devices are chosen.

In the following example, we consider the layout of a CMOS inverter, with lengths $L = 0.3 \text{ }\mu\text{m}$ and widths $Wn = 1.5 \text{ }\mu\text{m}$ and $Wp = 4.5 \text{ }\mu\text{m}$.

In the example we use Magic and work with the iic-osic-tools Docker from JKU.



Magic: GUI

www.opencircuitdesign.com/magic/

The screenshot shows the Magic GUI interface with several key components highlighted by red arrows and text labels:

- Console:** A window on the left displaying the command-line interface and execution logs. An arrow points to it from the label "Console".
- Grid options:** A menu item in the top bar, highlighted with a red box and an arrow pointing to it from the label "Grid options".
- DRC check:** A menu item in the top bar, highlighted with a red box and an arrow pointing to it from the label "DRC check".
- Information: technology, layer:** A menu item in the top bar, highlighted with a red box and an arrow pointing to it from the label "Information: technology, layer".
- Selection Box:** A small white square in the main workspace, with an arrow pointing to it from the label "Selection Box".
- Layers:** A vertical toolbar on the right side of the workspace, containing various patterned boxes. An arrow points to it from the label "Layers: left click to make layer visible, wheel to paint the box and right click to make layer invisible".

Magic: commands

Comandos:

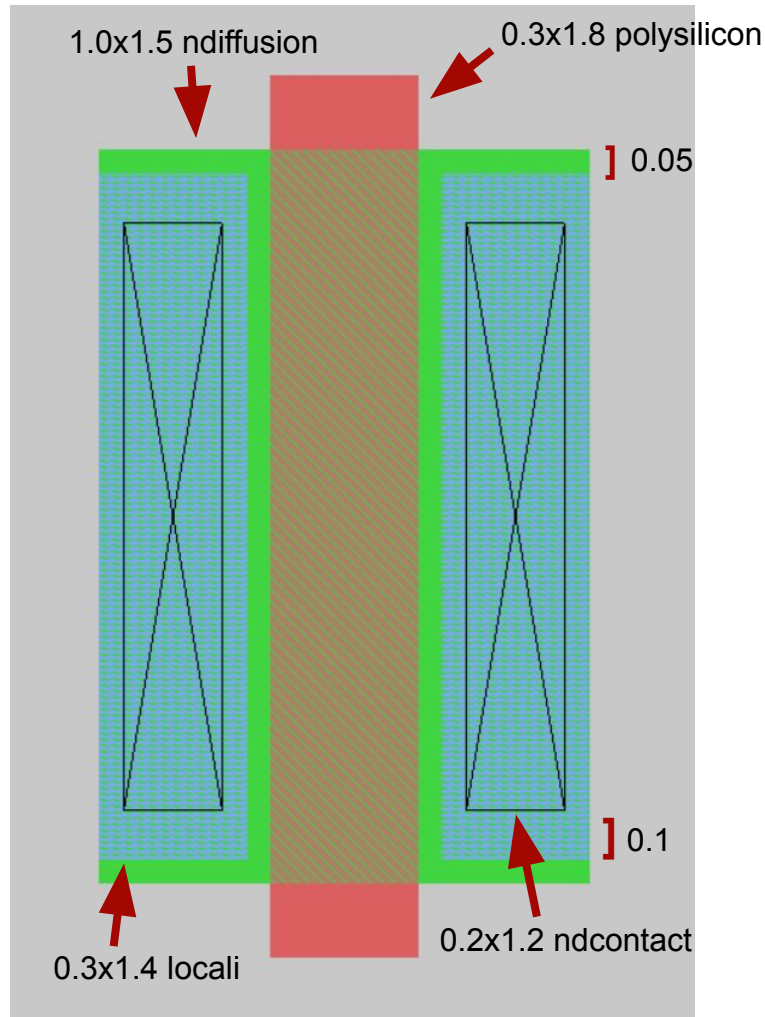
<i>box</i> <x1> <y1> <x2> <y2>	Choose box coordinates
<i>paint</i> <layer_name>	Paint box
<i>getcell</i> <cell_name>	Instantiate .mag file
<i>sideways</i>	Mirror horizontally
<i>upsidedown</i>	Mirror vertically
<i>rotate</i> [degrees]	Rotate (90° by default)
<i>label</i> <name> <dir> <layer>	Put label box
<i>source</i> <file_name>	Run .tcl script
<i>extract</i>	Extract devices
<i>ext2spice</i>	Convert extraction into .spice
<i>save</i> <cell_name>	Save .mag file
<i>load</i> <cell_name>	Load .mag file

left click	lower left corner of the box
right click	upper right corner of the box
'a'	select everything in the box
'b'	box information
'c'	copy
'd'	delete
's'	select polygon
'm'	move
'u'	undo
shift + 'u'	redo
'z'	zoom in
shift + 'z'	zoom out

Note: all the actions in the GUI can be executed with commands, and written in .tcl scripts

¿Which type is the substrate?

¿Where are the transistor terminals?



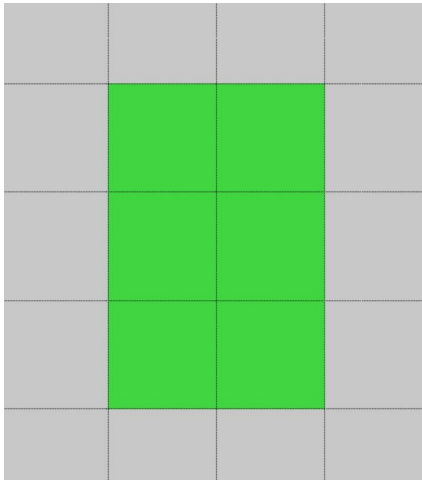
NMOS Layout

Execute magic and check technology in the upper right corner.

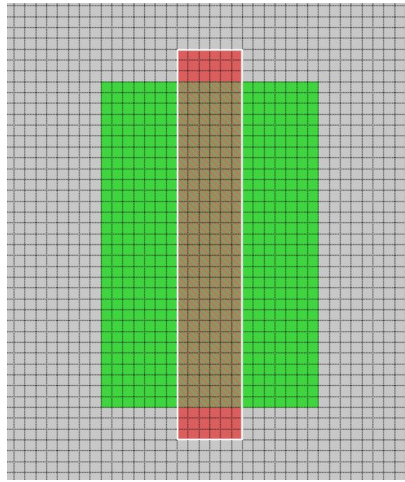
Draw the NMOS from the image, size $W=150\text{nm}$ and $L=30\text{nm}$

Rule		Value
difftap.1	diffusion min width	0.15um
difftap.3	diff. to diff. min spacing	0.27um
poly.1a	polysilicon min width	0.15um
poly.2	poly to poly min spacing	0.21um
poly.7	ndiffusion overhang of nFET	0.25um
difftap.2	Transistor min width	0.42um
poly.8	min poly overhang of FET	0.13um

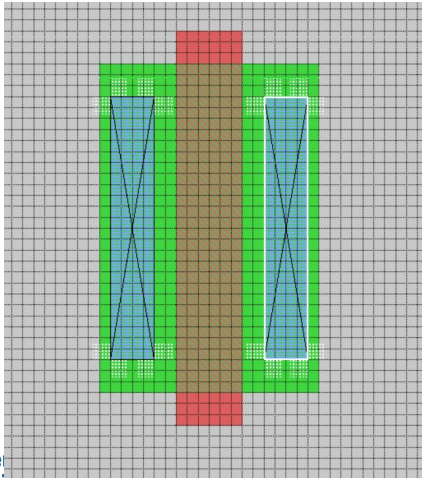
1.



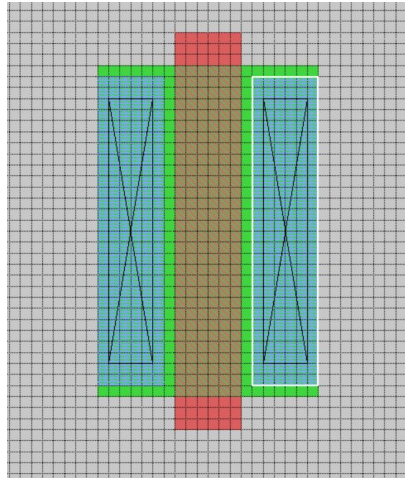
2.



3.



4.



On the *window* menu, select a 0.5um grid and enable *snap to grid*. Paint a 1.0um*1.5um rectangle width *ndiffusion* layer.

Change the grid size to 0.05 and paint a 0.3um*1.8um rectangle with *polysilicon*.

This will generate a 0.3um*1.5um *ntransistor*.

Paint a 0.2um*1.2um rectangle with *ndcontact* on each side.

Paint a 0.3um*1.4um rectangle with *locali* on each side.

Check DRC

PMOS Layout

A PMOS transistor has a similar construction to the NMOS, except it is inside an nwell and diffusions are p+ type. The required layers are:

- nwell
- pdiffusion
- polysilicon
- pdcontact
- locali

For the example inverter we need a PMOS with $W=4.5\mu\text{m}$ and $L=0.3\mu\text{m}$. What about aspect ratio?

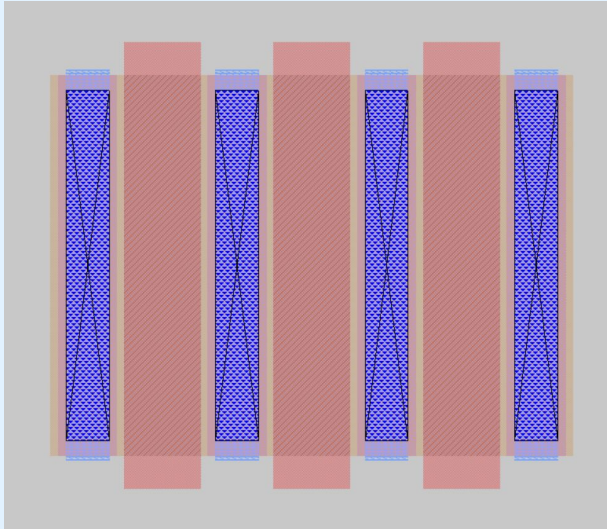
Scripting

Remembering that all actions can be done by commands, .tcl scripts can be used to do layout that is easily reproduced and changed.

```
# PMOS
box 0.2um 2.7um 2.6um 4.2um
paint pdiffusion
box 0.55um 2.5um 0.85um 4.35um
paint polysilicon
box 1.25um 2.5um 1.55um 4.35um
paint polysilicon
box 1.95um 2.5um 2.25um 4.35um
paint polysilicon
box 0.25um 2.85um 0.45um 4.05um
paint pdcontact
box 0.95um 2.85um 1.15um 4.05um
paint pdcontact
box 1.65um 2.85um 1.85um 4.05um
paint pdcontact
box 2.35um 2.85um 2.55um 4.05um
paint pdcontact
```


Device Generator

It is not necessary to do layout manually for each device.



params

New device: sky130_fd_pr_pfet_01v8 Library: sky130
Cellname: (default) Instance: (default)

Width (um) 1.5
Length (um) 0.15
Fingers 3
M 1
Device type sky130_fd_pr_pfet_01v8

Diffusion contact coverage (%) 100
Poly contact coverage (%) 100
Guard ring contact coverage (%) 100
Guard ring top/bottom contact coverage (%) 100

Overlap at poly contact
Overlap at diffusion contact
Add top gate contact
Add bottom gate contact
Add guard ring
Full metal guard ring
Add left guard ring contact
Add right guard ring contact
Add bottom guard ring contact
Add top guard ring contact

Source via coverage [+/-](%) 100
Drain via coverage [+/-](%) 100
Gate via coverage [+/-](%) 100
Bottom guard ring via coverage [+/-](%) 0
Top guard ring via coverage [+/-](%) 0
Right guard ring via coverage [+/-](%) 0
Left guard ring via coverage [+/-](%) 0

Create Create and Close Reset Close

Layout v/s Schematic

Layout v/s Schematic (LVS)

In Xschem:

- Generate .spice by pressing 'Netlist'
- Open netlist with 'Simulation>Edit Netlist'
- Save .spice in work directory

```
** sch_path: /foss/designs/Talleres/not.sch
**_subckt not VDD GND IN OUT
*.iopin VDD
*.iopin GND
*.iopin IN
*.opin OUT
XM1 OUT IN GND GND sky130_fd_pr_nfet_01v8 L=0.3 W=1.5 nf=1 ad='int((nf+1)/2) * W/nf * 0.29' as='int((nf+2)/2) * W/nf * 0.29'
+ pd='2*int((nf+1)/2) * (W/nf + 0.29)' ps='2*int((nf+2)/2) * (W/nf + 0.29)' nrd='0.29 / W' nrs='0.29 / W'
+ sa=0 sb=0 sd=0 mult=1 m=1
XM2 OUT IN VDD VDD sky130_fd_pr_pfet_01v8 L=0.3 W=4.5 nf=1 ad='int((nf+1)/2) * W/nf * 0.29' as='int((nf+2)/2) * W/nf * 0.29'
+ pd='2*int((nf+1)/2) * (W/nf + 0.29)' ps='2*int((nf+2)/2) * (W/nf + 0.29)' nrd='0.29 / W' nrs='0.29 / W'
**_ends
.ends
```

In Magic:

- Extract with: *extract*
- Configure with: *ext2spice lvs*
- Configure with: *ext2spice subcircuit top off*
- Get .spice: *ext2spice*

```
* SPICE3 file created from inverter.ext - technology: sky130A
X0 Vdd Vin Vout Vdd sky130_fd_pr_pfet_01v8 ad=0.3 pd=1.9 as=0.3 ps=1.9 w=1.5 l=0.3
X1 Vout Vin Gnd Gnd sky130_fd_pr_nfet_01v8 ad=0.525 pd=3.7 as=0.525 ps=3.7 w=1.5 l=0.3
X2 Vout Vin Vdd Vdd sky130_fd_pr_pfet_01v8 ad=0.3 pd=1.9 as=0.525 ps=3.7 w=1.5 l=0.3
X3 Vout Vin Vdd Vdd sky130_fd_pr_pfet_01v8 ad=0.525 pd=3.7 as=0.3 ps=1.9 w=1.5 l=0.3
```

Once both netlists are in the same directory, execute netgen

netgen -batch lvs net1.spice net2.spice \$PDK_PATH/libs.tech/netgen/sky130A_setup.tcl lvs.log

LVS log

<http://opencircuitdesign.com/netgen/tutorial/tutorial.html#Results>

```
Class inverter_layout.spice (1): Merged 2 parallel devices.
```

```
Subcircuit summary:
```

```
Circuit 1: inverter_schem.spice
```

```
|Circuit 2: inverter_layout.spice
```

```
-----|-----  
sky130_fd_pr__nfet_01v8 (1)
```

```
|sky130_fd_pr__nfet_01v8 (1)
```

```
sky130_fd_pr__pfet_01v8 (1)
```

```
|sky130_fd_pr__pfet_01v8 (3->1)
```

```
Number of devices: 2
```

```
|Number of devices: 2
```

```
Number of nets: 4
```

```
|Number of nets: 4  
-----|-----
```

```
Netlists match uniquely.
```

```
Cells have no pins; pin matching not needed.
```

```
Device classes inverter_schem.spice and inverter_layout.spice are equivalent.
```

```
Final result: Circuits match uniquely.
```

The background of the slide is a detailed, top-down view of a microchip layout simulation. It features a complex grid of rectangular blocks, lines, and patterns in various colors including green, blue, brown, and grey, all set against a dark teal background. The layout is dense and intricate, representing the physical design of a semiconductor device.

Post-Layout Simulation

Parasitic extraction

Netlist for LVS

extract

ext2spice cthresh 0

ext2spice rthresh 0

ext2spice

```
* NGSPICE file created from inverter.ext - technology: sky130A
.subckt inverter
X0 Vdd Vin Vout Vdd sky130_fd_pr__pfet_01v8 ad=0.3 pd=1.9 as=0.3 ps=1.9 w=1.5 l=0.3
X1 Vout Vin Gnd Gnd sky130_fd_pr__nfet_01v8 ad=0.525 pd=3.7 as=0.525 ps=3.7 w=1.5 l=0.3
X2 Vout Vin Vdd Vdd sky130_fd_pr__pfet_01v8 ad=0.3 pd=1.9 as=0.525 ps=3.7 w=1.5 l=0.3
X3 Vout Vin Vdd Vdd sky130_fd_pr__pfet_01v8 ad=0.525 pd=3.7 as=0.3 ps=1.9 w=1.5 l=0.3
.ends
```

Netlist with parasitics

```
X0 Vdd Vin Vout Vdd sky130_fd_pr__pfet_01v8 ad=0.3 pd=1.9 as=0.3 ps=1.9 w=1.5 l=0.3
X1 Vout Vin Gnd Gnd sky130_fd_pr__nfet_01v8 ad=0.525 pd=3.7 as=0.525 ps=3.7 w=1.5 l=0.3
X2 Vout Vin Vdd Vdd sky130_fd_pr__pfet_01v8 ad=0.3 pd=1.9 as=0.525 ps=3.7 w=1.5 l=0.3
X3 Vout Vin Vdd Vdd sky130_fd_pr__pfet_01v8 ad=0.525 pd=3.7 as=0.3 ps=1.9 w=1.5 l=0.3
C0 Vin Vdd 0.332f
C1 Vout Vdd 0.307f
C2 Vout Vin 0.225f
C3 Vout Gnd 0.301f
C4 Vin Gnd 0.513f
C5 Vdd Gnd 0.979f
.end
```