

# **Layout-Based Design and Verification of CMOS Standard Cells**

Using SCMOS Technology in Magic

**GUNJA JASWANTH**

B.Tech Electrical and Computer Engineering  
Shiv Nadar University

# Contents

<b>1 Abstract</b>	<b>3</b>
<b>2 SCMOS Technology Overview</b>	<b>3</b>
<b>3 Project Flow</b>	<b>3</b>
<b>4 CMOS Inverter</b>	<b>4</b>
4.1 Layout . . . . .	4
4.2 Truth Table . . . . .	4
4.3 SPICE Waveforms . . . . .	5
4.4 IRSIM Commands Used for Inverter . . . . .	5
4.5 IRSIM Logical Waveform . . . . .	6
<b>5 2-Input NAND Gate</b>	<b>6</b>
5.1 Layout . . . . .	6
5.2 Truth Table . . . . .	6
5.3 SPICE Waveforms . . . . .	7
5.4 IRSIM Commands Used . . . . .	8
5.5 IRSIM Logical Waveform . . . . .	9
<b>6 2-Input NOR Gate</b>	<b>9</b>
6.1 Layout . . . . .	9
6.2 Truth Table . . . . .	9
6.3 SPICE Waveforms . . . . .	10
6.4 IRSIM Commands Used . . . . .	11
6.5 IRSIM Logical Waveform . . . . .	11
<b>7 Learning Outcomes</b>	<b>12</b>
<b>8 Conclusion</b>	<b>12</b>

# 1 Abstract

This project presents the layout-first implementation and verification of CMOS standard cells including Inverter (INV), 2-input NAND, and 2-input NOR using SCMS technology rules in Magic.

The implemented flow:

1. Layout creation in Magic
2. DRC verification
3. Netlist extraction
4. Post-layout SPICE simulation
5. Logical verification using IRSIM

# 2 SCMS Technology Overview

SCMS (Scalable CMOS) is a lambda-based scalable layout rule system where geometries are defined in terms of  $\lambda$ .

Layers used in layout:

- N-well (PMOS region)
- Diffusion (Active)
- Polysilicon (Gate)
- Metal1 (Routing)
- Contacts (Poly-to-Metal and Diffusion-to-Metal)

Power rails (VDD and GND) were implemented using Metal1 for proper connectivity and routing. A MOS transistor is formed when polysilicon crosses diffusion.

# 3 Project Flow

Commands used in Magic:

```
drc check  
extract all  
ext2spice  
ext2sim
```

After extraction, the generated SPICE file was simulated in NGSPICE. IRSIM was used for logical verification.

## 4 CMOS Inverter

### 4.1 Layout

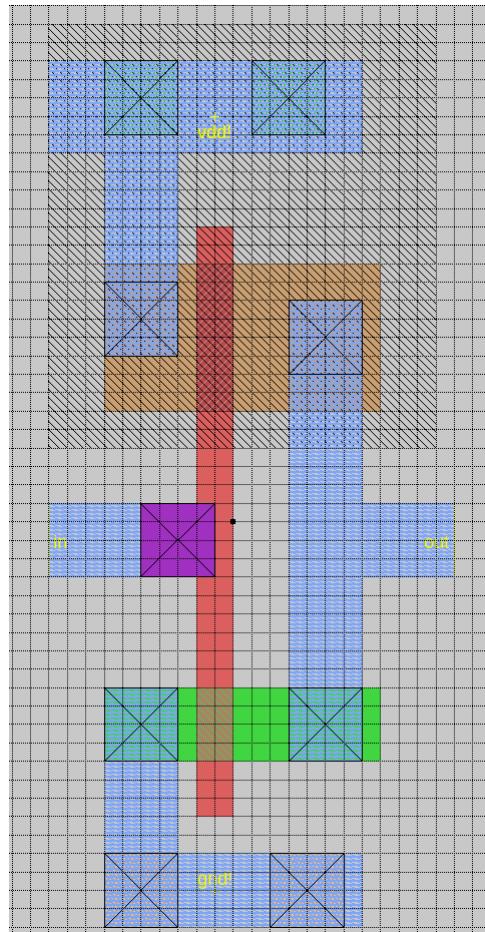


Figure 1: Inverter Layout

### 4.2 Truth Table

Input	Output
0	1
1	0

### 4.3 SPICE Waveforms

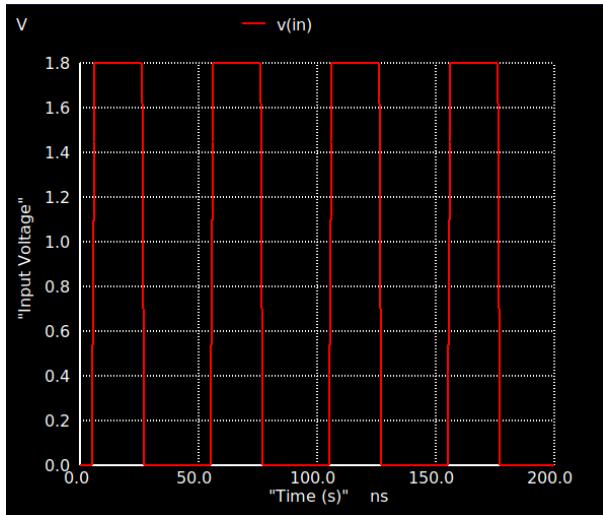


Figure 2: Input Waveform

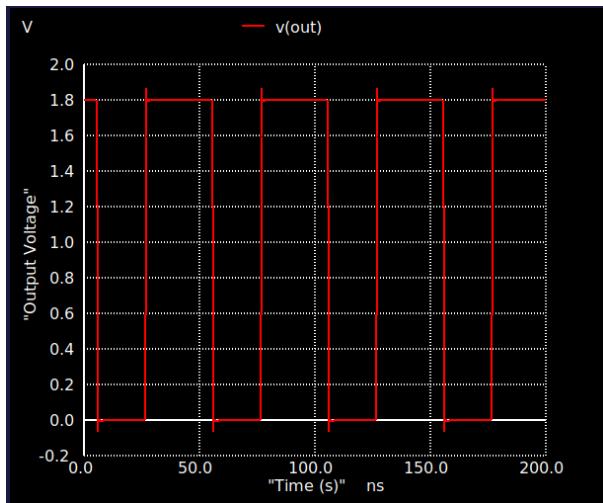


Figure 3: Output Waveform

### 4.4 IRSIM Commands Used for Inverter

The following commands were executed in IRSIM to verify the logical behavior of the CMOS inverter:

```
power vdd!
ground gnd!
h vdd!
l gnd!

analyzer in out

l in
```

s	
h in	
s	

## 4.5 IRSIM Logical Waveform



Figure 4: Inverter Logical Verification

## 5 2-Input NAND Gate

### 5.1 Layout

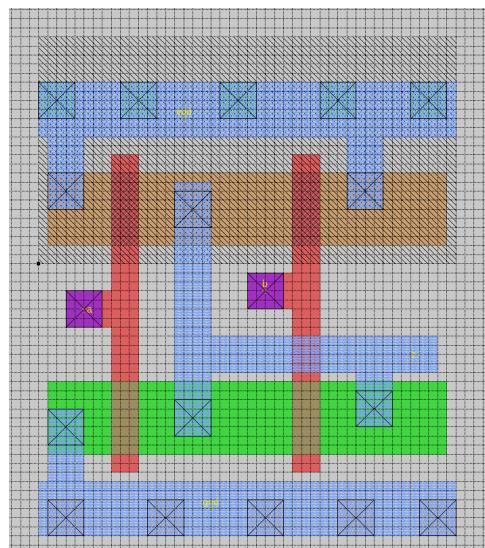


Figure 5: NAND Layout

### 5.2 Truth Table

A	b	z
0	0	1
0	1	1
1	0	1
1	1	0

### 5.3 SPICE Waveforms

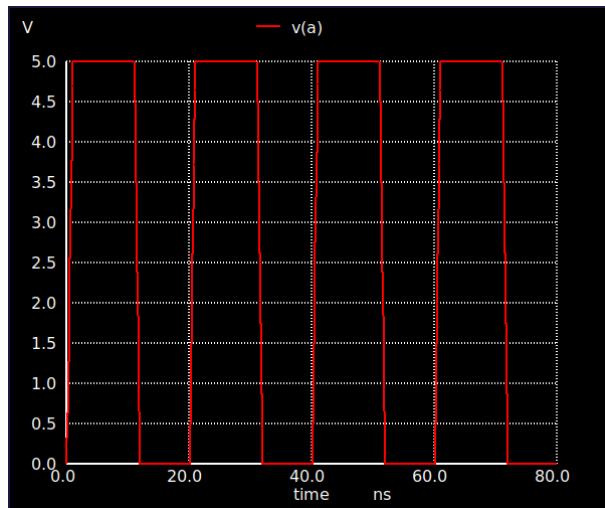


Figure 6: Input a

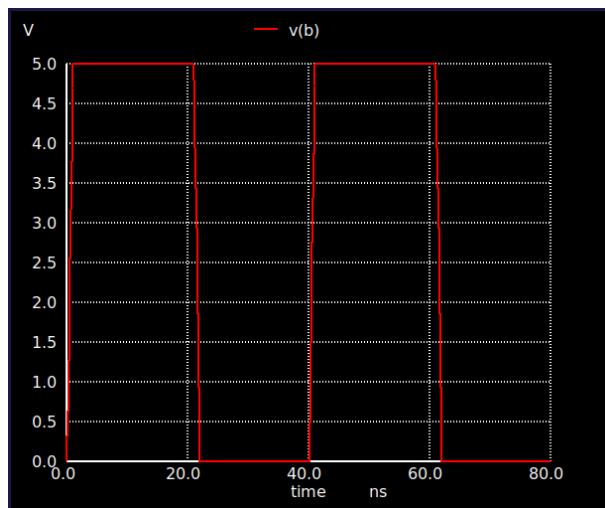


Figure 7: Input b

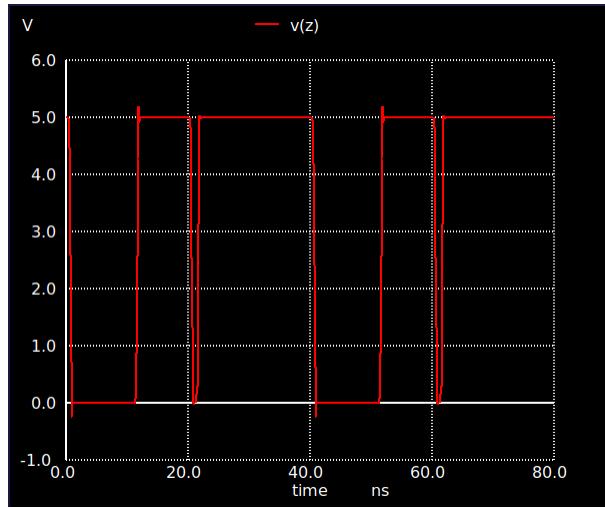


Figure 8: Output Waveform

#### 5.4 IRSIM Commands Used

The following IRSIM commands were executed to verify all input combinations of the gate:

```

power vdd
ground gnd
h vdd
l gnd

analyzer a b z

h a
h b
s

h a
l b
s

l a
l b
s

l a
h b
s

```

## 5.5 IRSIM Logical Waveform

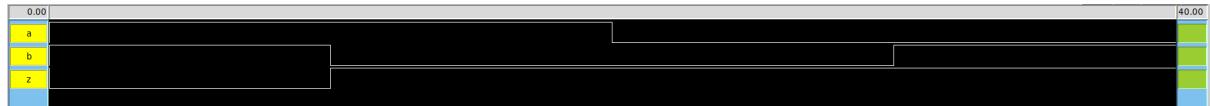


Figure 9: NAND Logical Verification

## 6 2-Input NOR Gate

### 6.1 Layout

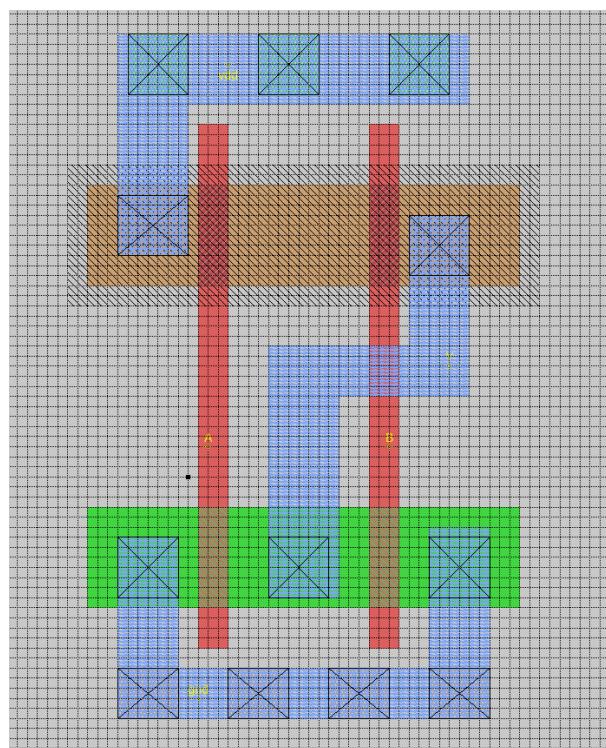


Figure 10: NOR Layout

### 6.2 Truth Table

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

### 6.3 SPICE Waveforms

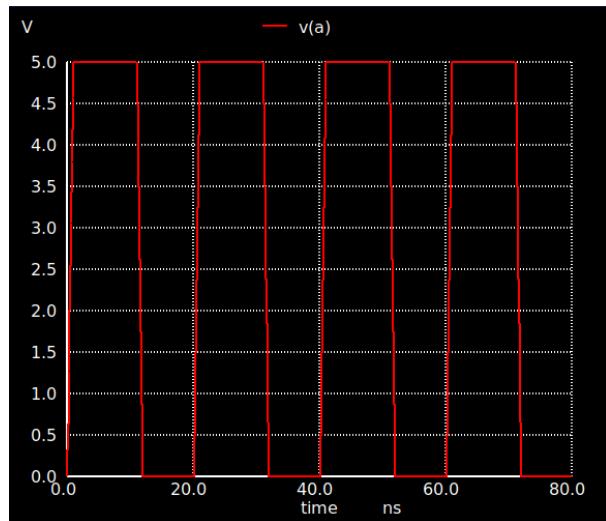


Figure 11: Input A

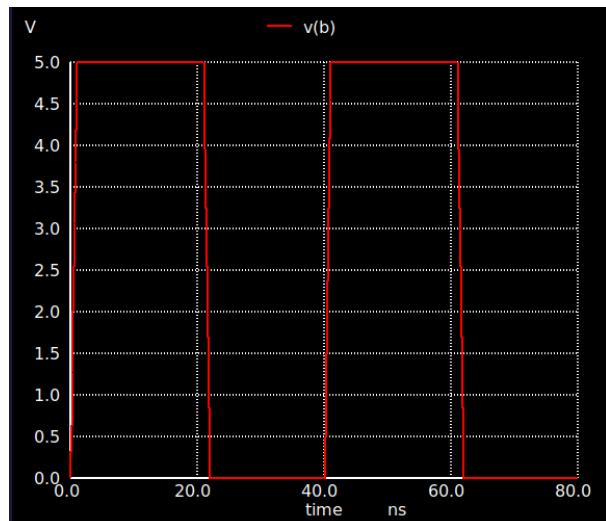


Figure 12: Input B

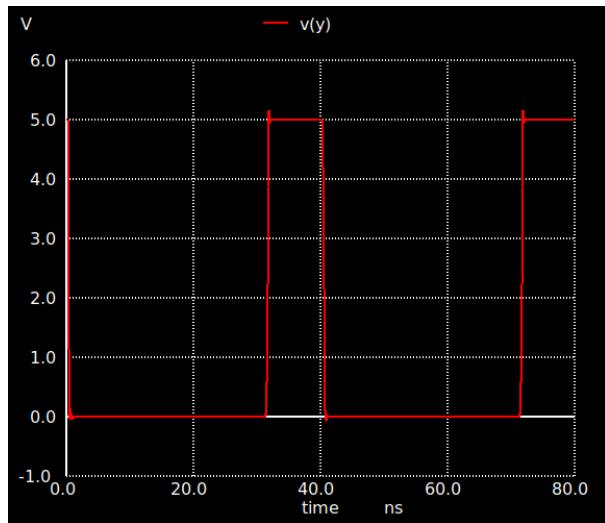


Figure 13: Output Waveform

## 6.4 IRSIM Commands Used

```

power vdd
ground gnd
h vdd
l gnd

analyzer A B Y

h A
h B
s

l A
l B
s

h A
l B
s

l A
h B
s

```

## 6.5 IRSIM Logical Waveform

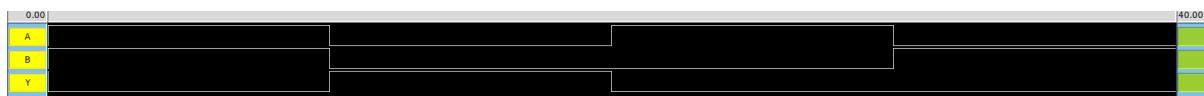


Figure 14: NOR Logical Verification

## 7 Learning Outcomes

- SCMOS lambda-based layout design
- Transistor creation using layer intersections
- Netlist extraction from layout
- Post-layout SPICE simulation
- Switch-level verification using IRSIM

## 8 Conclusion

The successful implementation and verification confirm the correctness of the layout-first methodology using SCOMS rules. The results validate that extracted netlists preserve logical functionality after physical layout realization.