# CMOS Design Project: 4 Bit Odd Parity Checker

**Indian Institute of Information Technology,**

**Nagpur**

**ECL 312: CMOS Design**

**A Project Report on: 4-Bit Odd parity checker using CMOS**

**Submitted By:**

**Aman Kaliramna (BT22ECE101)**

**Submitted to :**

Prof. Paritosh Peshwe

**Department of Electronics and Communication**

## Project Overview

This repository contains the design and simulation of a 4-bit odd parity checker implemented using CMOS technology. The project includes the complete circuit design, simulation using ngspice, and layout implementation using Microwind under 90nm technology. The odd parity checker is a fundamental component in error detection systems, ensuring data integrity by verifying whether the number of 1's in a 4-bit input is odd.

## Objectives

1. Design and simulate a 4-bit odd parity checker using CMOS technology.
2. Create and verify the **NgSpice netlist** for circuit simulation.
3. Implement the physical layout using **Microwind** in **90nm technology**.
4. Analyze and verify the circuit's performance through simulation results

## Design Description

### Circuit Operation

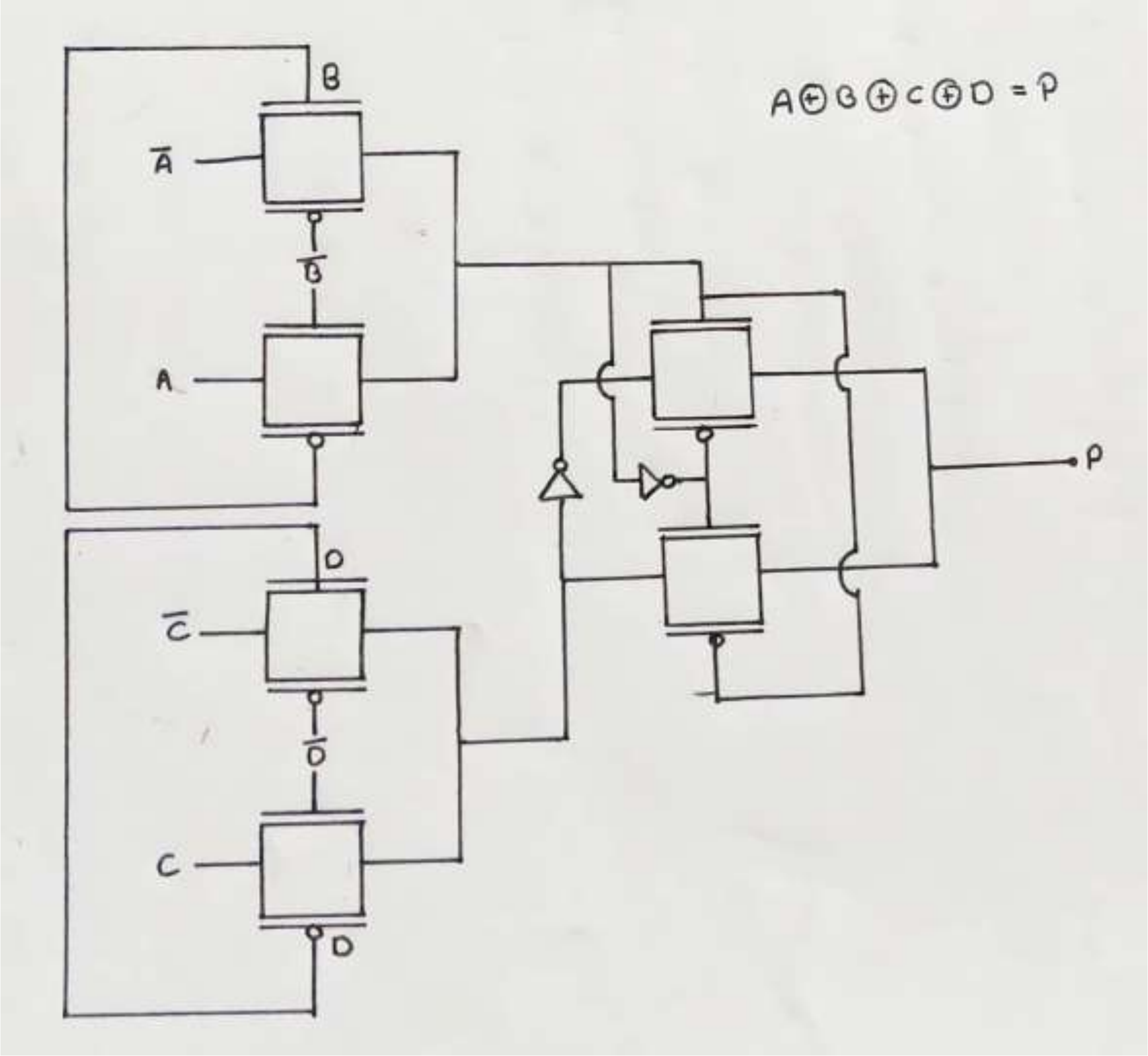
- Takes **4 input bits** (A, B, C, D)
- Produces a **parity bit** output (P)
- P can be compounded as $P=A\oplus B\oplus C\oplus D$ where $\oplus$ denotes XOR logic
- Output P = 0 when number of 1's in input is even (to make total number of 1's odd)
- Output P = 1 when number of 1's in input is odd

### Logic Implementation

The circuit implements the following logic:

- Uses **XOR gates** to check for odd/even number of 1's
- Implemented using **transmission gates** instead of conventional CMOS design for minimal area occupation.

## Transmission Gate Circuit:



## Simulation Files

Below are snippets of the circuit files used for simulation in **ngSpice**:

### ngspice circuit file:

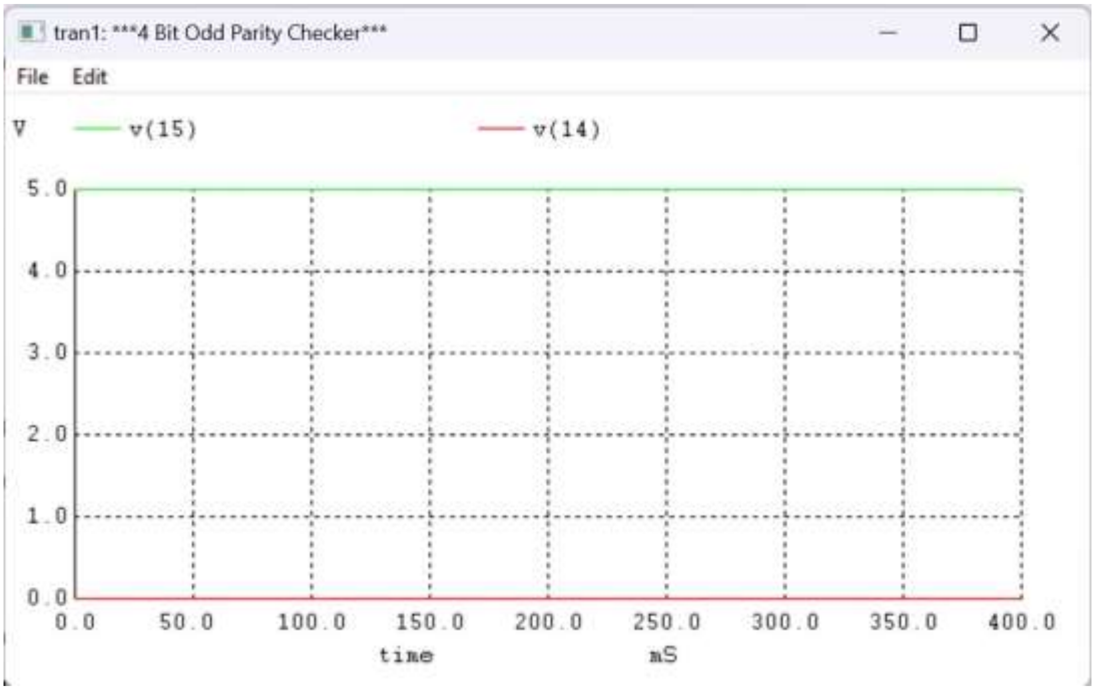
```
**4 Bit Odd Parity Checker**

.model nmod nmos level=54 version=4.7

.model pmod pmos level=54 version=4.7
```

```
.subckt inverter in vdd out
M1 out in 0 0 nmod w=100u l=10u
M2 out in vdd vdd pmod w=200u l=10u
.model nmod nmos level=54 version=4.7
.model pmod pmos level=54 version=4.7
.ends

.subckt tg in out sel selb
M1 out sel in in nmod w=100u l=10u
M2 out selb in in pmod w=200u l=10u
.ends

.subckt xor_gate a b out ab bb
Xab_b a out b bb tg
Xa_bb ab out bb b tg
.model nmod nmos level=54 version=4.7
.model pmod pmos level=54 version=4.7
.ends
Va 1 0 pulse(0 5 0 0 0 50m 100m)
Vb 2 0 pulse(0 5 0 0 0 100m 200m)
Vc 3 0 pulse(0 5 0 0 0 200m 400m)
Vd 4 0 pulse(0 5 0 0 0 400m 800m)
Vab 5 0 pulse(5 0 0 0 0 50m 100m)
Vbb 6 0 pulse(5 0 0 0 0 100m 200m)
Vcb 7 0 pulse(5 0 0 0 0 200m 400m)
Vdb 8 0 pulse(5 0 0 0 0 400m 800m)
```

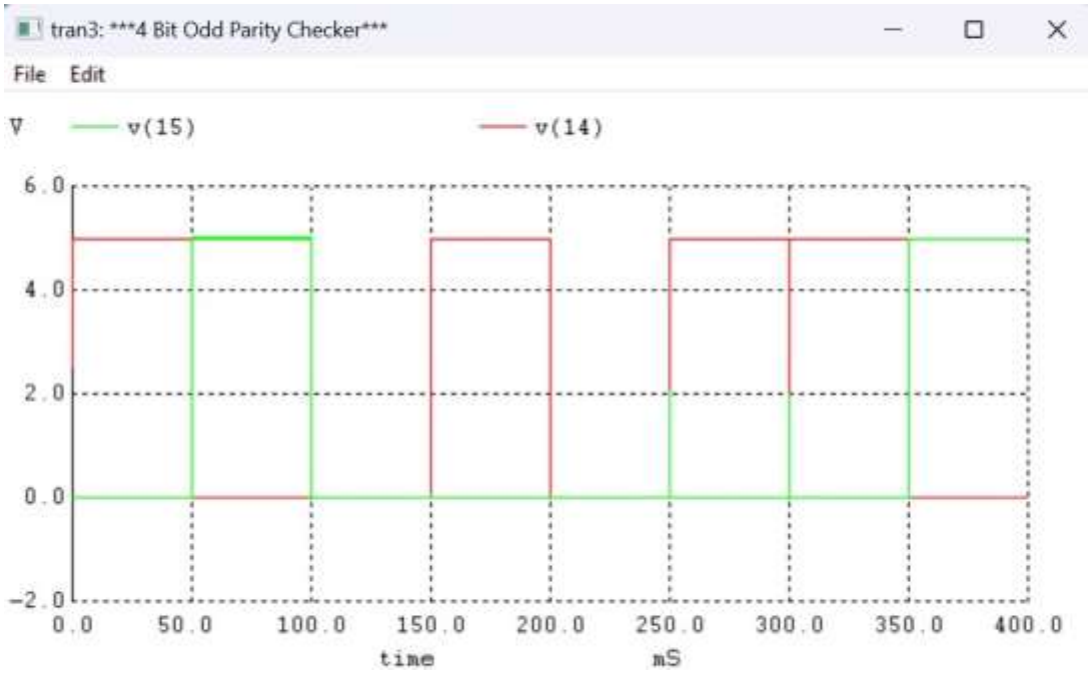
```
Vdd 9 0 dc 5

* For Testing the following value should output ON *
Va 1 0 dc 0v
Vb 2 0 dc 0v
Vc 3 0 dc 0v
Vd 4 0 dc 5v
Vab 5 0 dc 5v
Vbb 6 0 dc 5v
Vcb 7 0 dc 5v
Vdb 8 0 dc 0v

Xab 1 2 10 5 6 xor_gate
Xcd 3 4 11 7 8 xor_gate
Xab_inv 10 9 12 inverter
Xcd_inv 11 9 13 inverter
Xabcd 10 11 14 12 13 xor_gate
Xout 14 5 15 inverter

.tran 0.1m 400m
.control
run
plot v(1) v(2) v(3) v(4) v(14)
plot v(14) v(15)
.endc
.end
```

## Ngspice Output:

### For DC input:

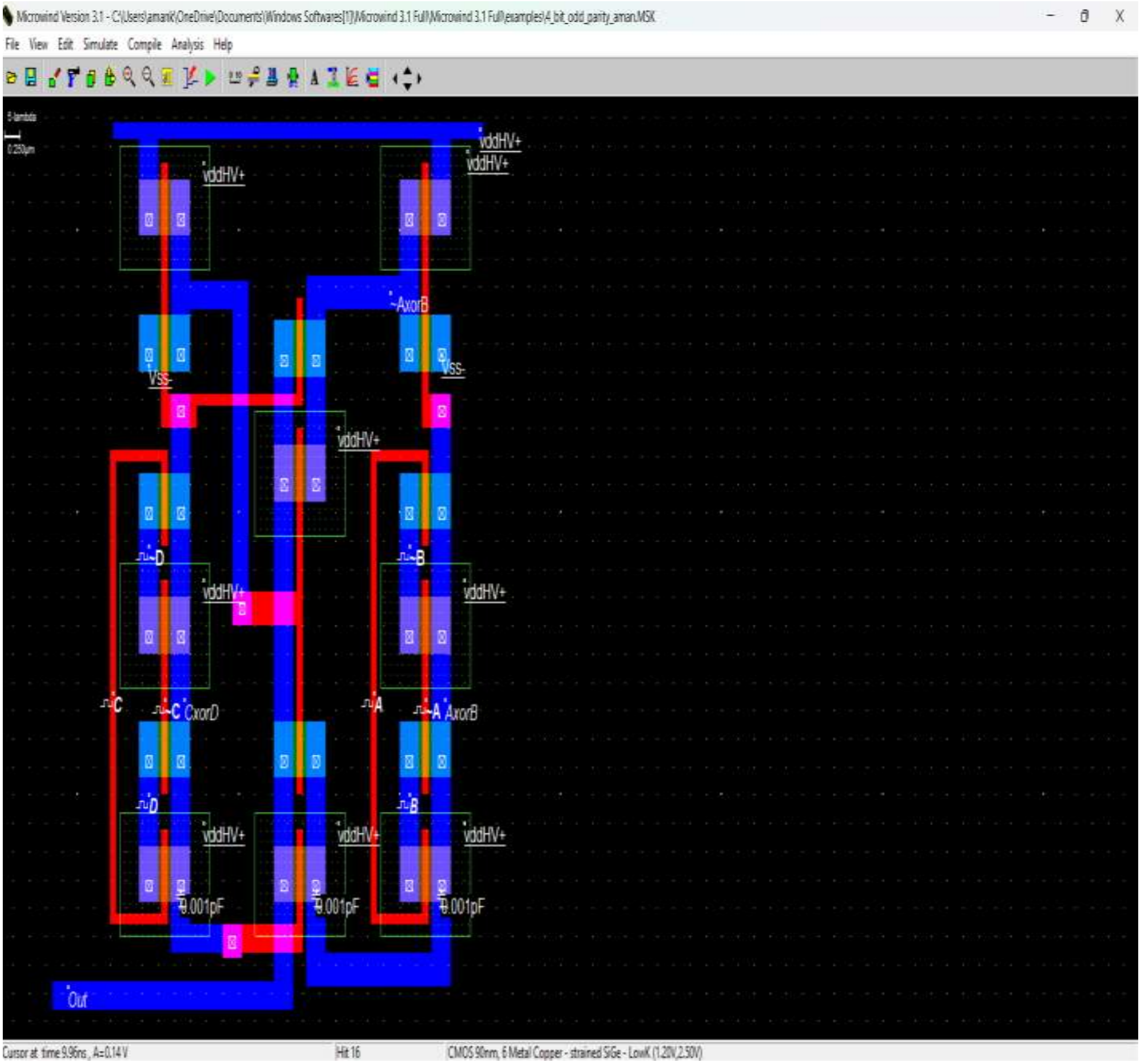


### For pulse input:



## Layout and Waveforms

## Circuit Layout:

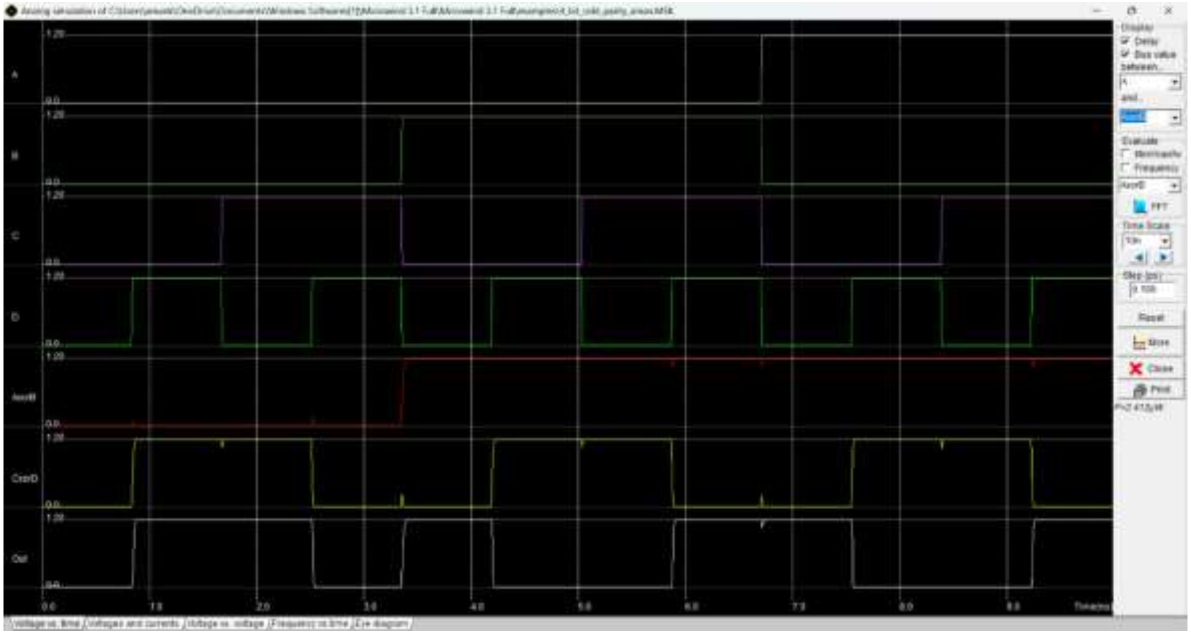


Cursor at time 9.96ns , A=0.14 V

Hit 16

CMOS 90nm, 6 Metal Copper - strained SiGe - LowK (1.20V,2.50V)

## Output Waveforms:



## Why Transmission Gates Over Pass Transistors?

The choice of transmission gates over pass transistors in this design was motivated by several key advantages:

- Voltage Level Preservation:
    - Pass transistors suffer from threshold voltage drop (NMOS passes weak '1', PMOS passes weak '0')
    - Transmission gates combine NMOS and PMOS transistors to ensure full voltage swing
    - This results in stronger logic levels and better noise margins
- Speed and Performance:
    - Pass transistors have asymmetric switching speeds (faster for one transition than the other)
    - Transmission gates provide balanced switching characteristics
    - Results in more predictable timing and better overall circuit performance
- Power Efficiency:

    - Pass transistors may require additional buffers to restore voltage levels
    - These buffers consume extra power and increase circuit complexity
    - Transmission gates eliminate the need for voltage restoration, reducing power consumption
- Reliability:
    - Pass transistors are more susceptible to noise and voltage variations
    - Transmission gates provide better immunity to noise
    - More reliable operation across different operating conditions
- Layout Considerations:
    - While transmission gates use more transistors than pass transistors
    - The elimination of voltage restoration circuitry often results in comparable or better area efficiency
    - Simpler integration with other circuit components

## Implementation Challenges and Solutions

**1. Signal Routing Complexity**

**Challenge:**

- Complex interconnections between multiple XOR gates visible in the layout
- Multiple crossing paths for signals A, B, C, and D
- Risk of signal interference due to dense routing in central area
- Need to maintain consistent delay across all input paths

**Solution:**

- Implemented a systematic routing strategy using different metal layers
- Used blue metal lines for main signal routing and red metal lines for cross connections
- Placed transmission gates in a symmetric pattern to minimize routing congestion

- Added strategic vias (marked with crosses in layout) to ensure proper layer connections
- Maintained equal path lengths for signals to avoid timing mismatches

**2. Output Signal Stability**

**Challenge:**

- Simulation waveforms show potential glitches during input transitions
- Multiple signal transitions need to be handled simultaneously
- Risk of incorrect parity output during simultaneous input changes
- Need for stable output across all 16 input combinations

**Solution:**

- Added 0.001pF capacitors (visible in layout) for output stabilization
- Implemented proper sizing of transmission gates (shown by gate dimensions in layout)
- Used balanced power distribution (vddHV+ routing) across the circuit
- Ensured proper ground (Vss-) connections for stable reference
- Verified functionality through comprehensive simulation (as shown in waveform)

## Future Work

Potential improvements and extensions could include:

1. Extending the design to handle 8-bit inputs
2. Optimizing the circuit for lower power consumption
3. Implementing error correction capabilities

## Conclusion

This project demonstrates the successful implementation of a 4-bit odd parity checker using CMOS technology. The design achieves:

- Correct functionality verified through simulation
- Implementation in 90nm technology
- Use of transmission gates