|  |  |
| --- | --- |
| Half adder uSING DOMINO LOGIC |  |
| DIGITAL VSLI DESIGN |

Shrieya D R

SRN: PES2UG22EC134

Section: 4B

Department of Electronics & Communication

Date: April 25, 2024

Implementation of a Half Adder using Domino Logic

-> in the cadence software

# Abstract

# Introduction

# Design and Implementation

# Simulation and Testing

# Conclusion

# References

# Acknowledgement

# Abstract

This project focuses on the hierarchical implementation of a half adder using domino logic

The software used for the project is: Cadence

The simulator given as the primary backbone for Quartus software to run: Spectre.

The motivation lies in exploring modular design principles in digital circuits to achieve a scalable and efficient addition operation. The project involves the design and implementation of half adder using domino logic; Alongside calculations of delay with the transient analysis of the circuit.

The Schematic for the same is generated, and simulations are conducted to validate the correctness of the design.

Performance metrics such as propagation delay, power consumption, and area efficiency are evaluated to demonstrate the effectiveness of the domino logic approach in implementing the half adder circuit.

This project contributes to the understanding of digital circuit design by exploring the application of domino logic in implementing fundamental arithmetic circuits, specifically here, the half adder and serves as a foundational exploration for more complex arithmetic logic units.

# Introduction

A half-adder has two inputs, A and B, and two outputs, S and Cout.

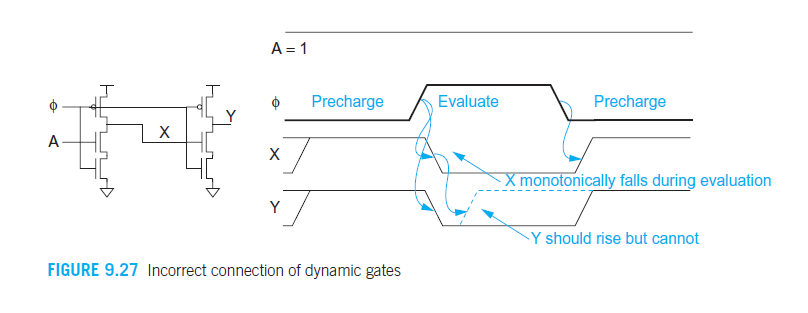
S is the sum of A and B. If A and B are both 1, S is 2, which cannot be represented with a single binary digit. So, it is indicated with a carry out Cout in the next column.

The half adder can be built from an XOR gate and an AND gate.

The issue addressed by Domino logic:

For Dynamic logic circuits, the inputs must be monotonically rising for the dynamic gate to compute the correct function.

However, the output of a dynamic gate begins HIGH and monotonically falls low during evaluation.



Dynamic gates sharing the same clock cannot be directly connected. This problem is often overcome with domino logic.

Additionally for dynamic logic, when the clock signal=0, Data=1; output may not be computed leading to a fail case hence it always requires its first clock input to be 1.

For the case of CLK=1 and Data=0; discharge is not a complete cycle and floating values will exist in such cases.

The Domino Logic:

Inserting a CMOS inverter between dynamic gates converts the monotonically falling output into a monotonically rising signal suitable for the next gate. The dynamic-static pair together is called a domino gate.

It needs just K+2 transistors to be built where K= no.of inputs at either the pull up or pull-down networks.

The ‘2’ is due to one NMOS clock and one PMOS clock.

Disadvantages:

1. Race conditions:

If signals arrive at gates too closely together, it can lead to incorrect logic states and unpredictable behavior.

1. Power & Energy consumption

Domino logic circuits typically consume more power compared to static CMOS circuits, due to the dynamic charging and discharging of nodes in the logic gates, which can lead to increased energy dissipation.

# Design and Implementation

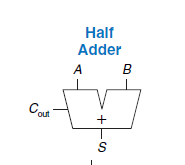
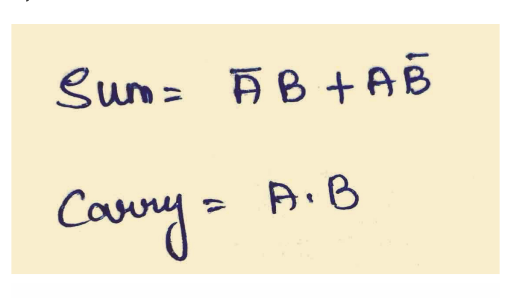
**Half Adder:**

The 1-bit Half adder is designed as per the logic given (here) in the truth table for a full adder.

|  |  |  |  |
| --- | --- | --- | --- |
| Input a | Input b | Sum (s) | Carryout (Cout) |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Step1:

Sum and Carryout (cout) equations can be formed as below by looking into the above-mentioned truth table of the full-adder.

For Domino logic, we take either pull up network (PMOS) or pull down network(NMOS) alone.

Here, considering NMOS, (the pull down network)

The pull up network has been replaced by a single PMOS transistor

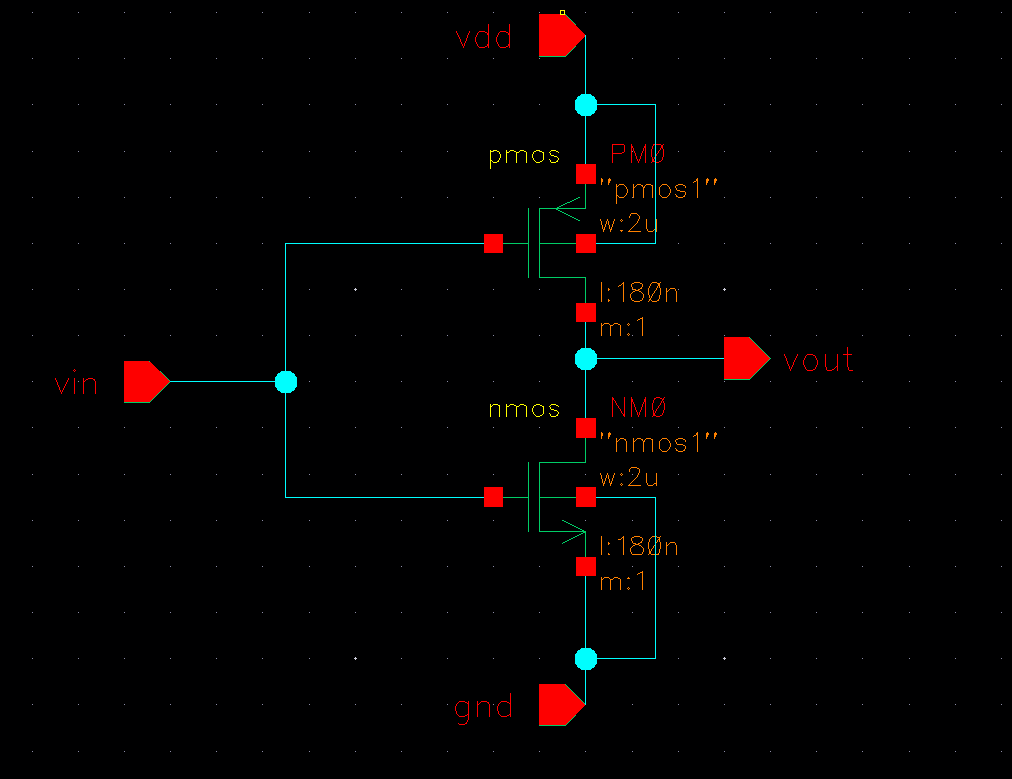
Below the Pull down network, an NMOS transistor has been placed.

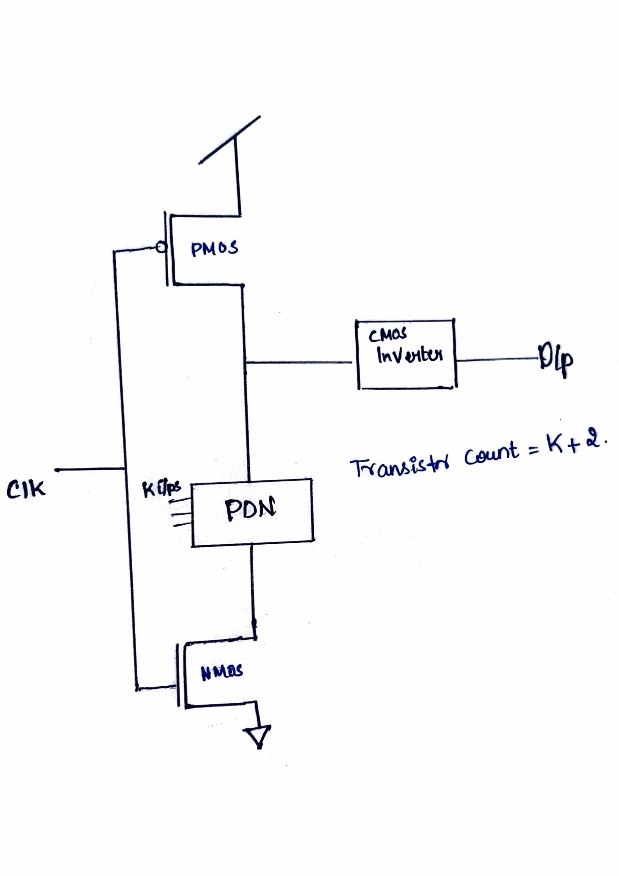
Both these transistors have been triggered by the same input clock signal. i.e., the gated clock

CMOS inverter has been added at the output

The schematic is as follows:

**CMOS SCHEMATIC:**



The domino logic circuit:

Half adder with domino logic sum schematic:

As mentioned earlier, only one network will be considered, here PDN [NMOS] is used

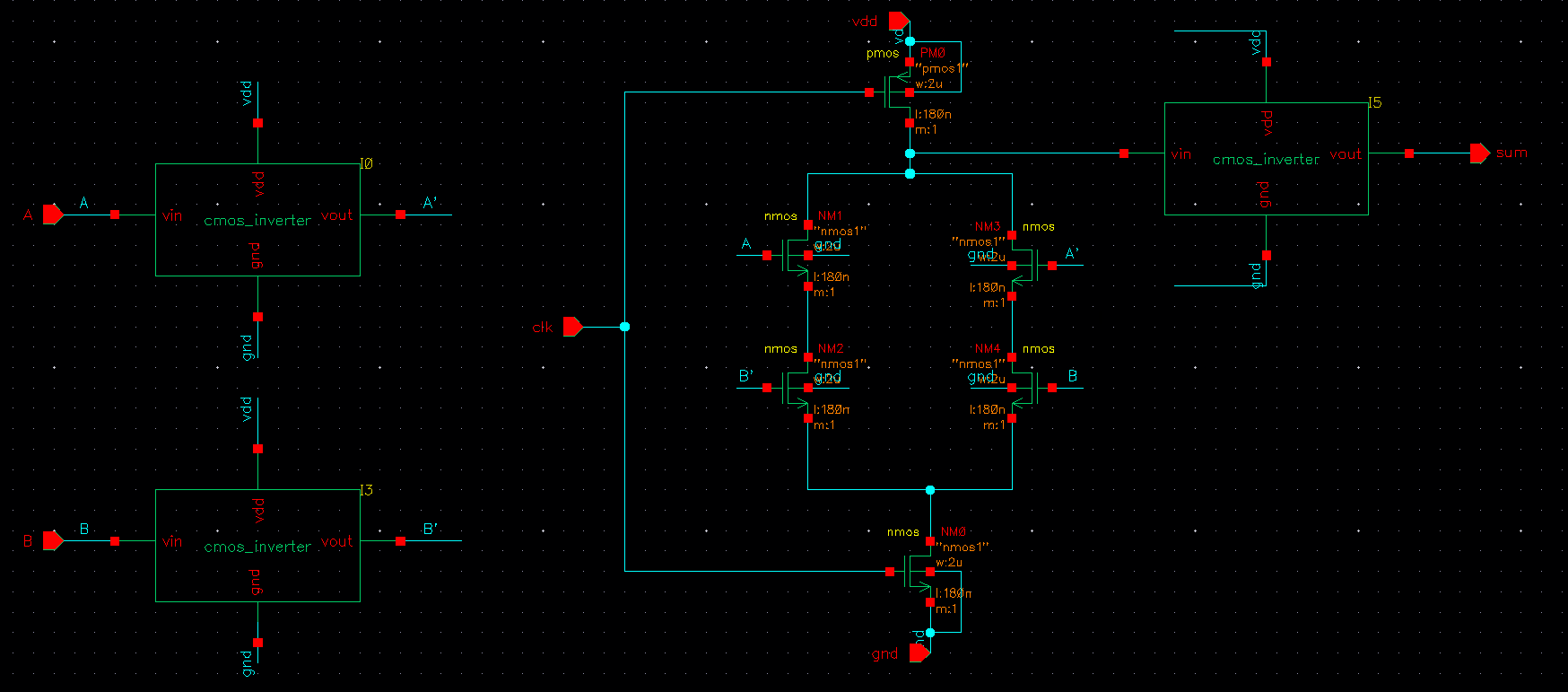
A file with the schematic of the circuit and just the input and output pins are provided.

Further, a symbol for the same schematic is created

By clicking on create -> Cellview -> from Cellview.

And a file for test is created where test voltages are applied to this symbol to test the circuit.

**THE SCHEMATIC FILES:**

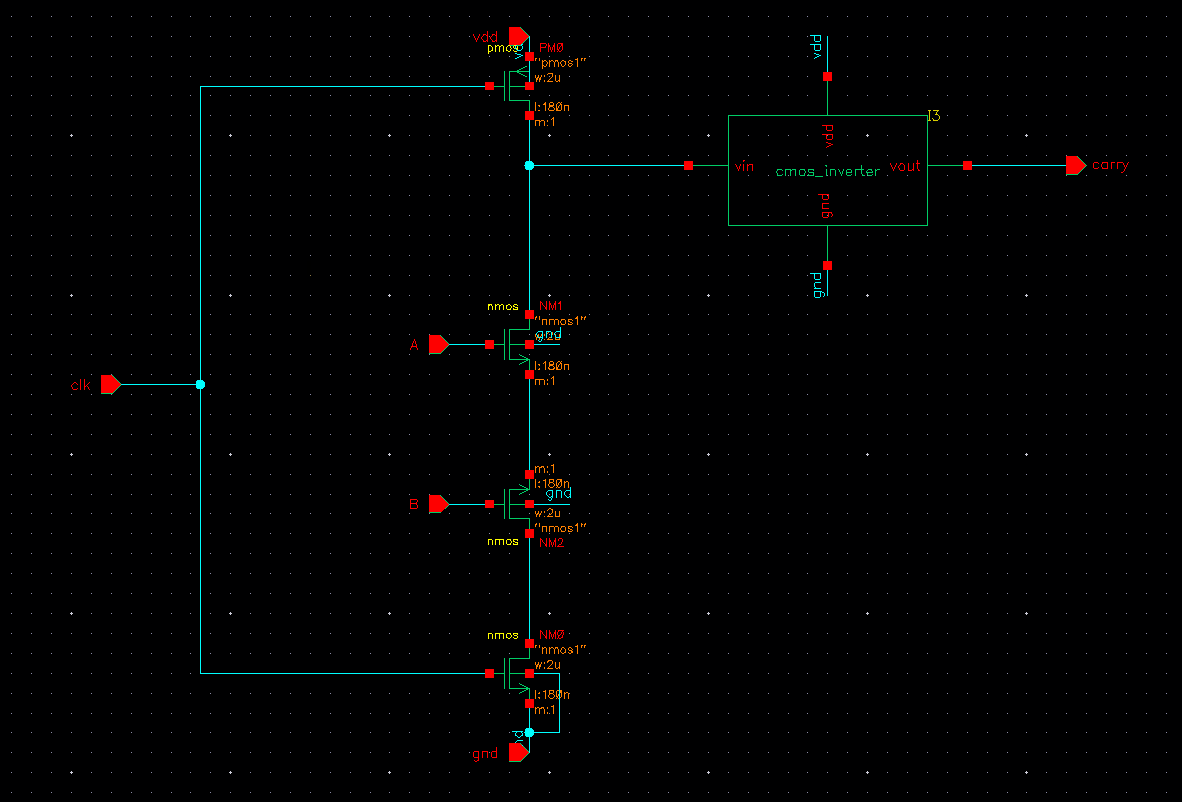
1. SUM

The sum logic for a half adder houses an XOR gate logic, when the inputs are different, output=1;Since only 2 pins of A and B must be considered for the simulation , CMOS inverters that were initially created in the software, have been converted into a symbol and extracted here.

Input A has been given to the CMOS inverter to invert A as A’ ; Input B has been given to the CMOS inverter to invert B as B’ & the inputs have been tapped accordingly . With the same clk signals for the PMOS and NMOS

Q=AB’+A’B ; for a Pull-Down Network, Multiplication of inputs must be connected in series and Addition of inputs must be connected in parallel. A CMOS inverter has been placed before the output pin to get rid of the ‘Z’ values [floating values] due to which the circuit might not discharge completely and would float in between.

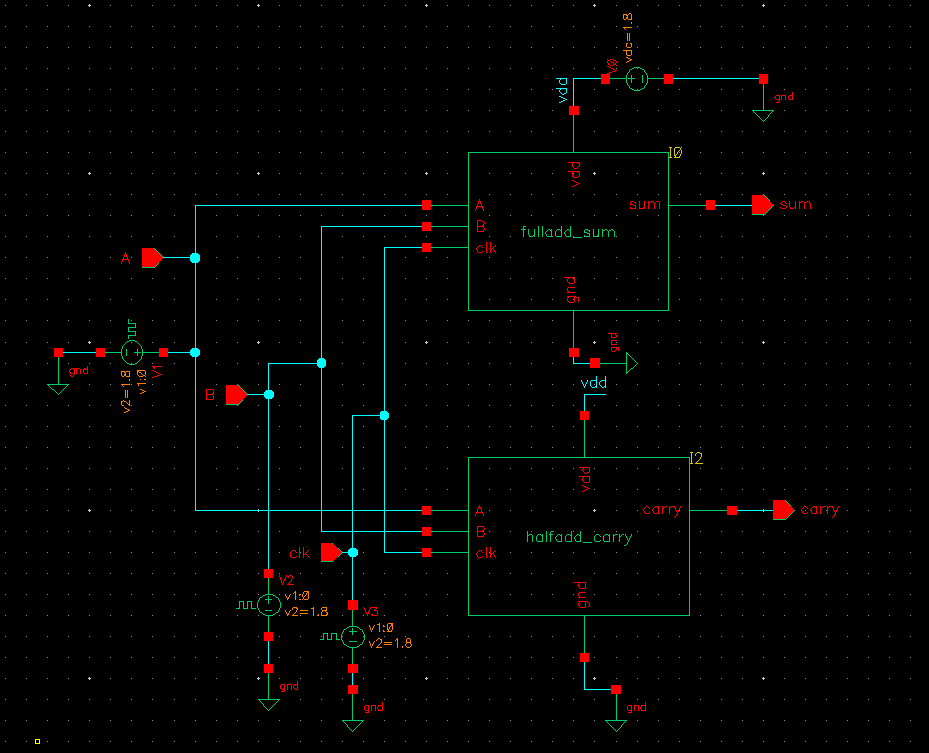
1. Carry



The carry output of a half adder circuit is a crucial element in binary addition, indicating whether there is a need to carry over to the next higher significant bit. The carry logic for a half adder houses an AND gate logic, only when both the inputs = 1, output=1; signifying the need to carry over to the next bit. When any one of the inputs=0; the output=0; indicating that there is no need of carry over to next bit

Since the circuit can be designed in a way in the pull down network ; multiplication in series; due to which the input pins have been tapped at both the transistors that have been placed in series and the output has been tapped after the PDN, and a CMOS inverter has been placed before the output pin to get rid of the floating values that might occur at the output leading to incomplete discharge or incomplete outputs as the output might float and this issue has been addressed by connecting a CMOS inverter which is capable of pushing any floating values to either VOH or VOL.

**Half Adder Logic**



Both the schematic files of carry and sum have been saved and a symbol has been created for each and a new file for the half-adder is created

And in the instance; library where sum and carry symbols have been housed have been extracted and placed and the corresponding voltages for the input pins have been provided as follows:

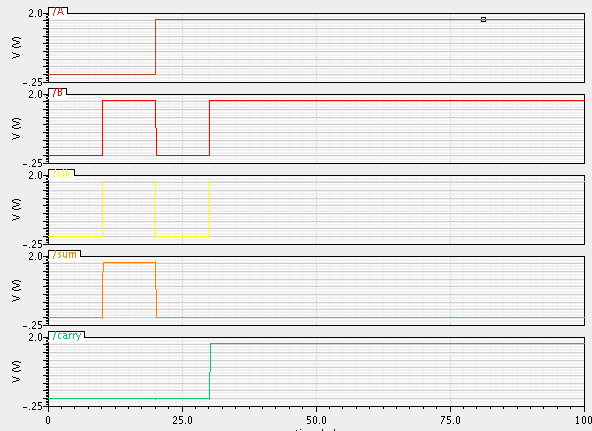
For input A:

Vbit: Pattern Parameter data: 0011 with a delay and fall time of 10pico seconds and period being 10nano seconds.

For input B:

Vbit: Pattern Parameter data: 0101 with a delay and fall time of 10pico seconds and period being 10nano seconds.

**Simulation and Testing**



Transient Response – Domino Logic

As one can observe here ,

Case1: At the input A =0; B=0; clk=0; the system retains the state and the output of both sum and carry=0 irrespective of the input signals

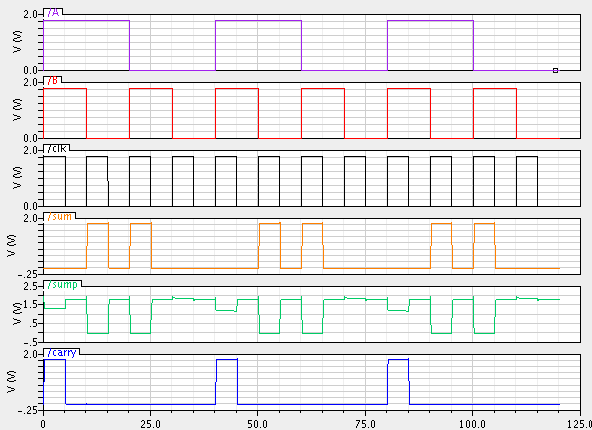
Case2: At the input A=0; B=1; clk=1; the system is expected to have the sum =1 and carry=0 as the sum follows XOR gate logic and carry follows AND logic which is satisfied here as the output -> sum=1; carry=0.

Case3: clk=0 due to which both sum and carry=0

Case4: At the input A=1; B=1; as per XOR logic the expected sum output = 0; expected carry output as per AND logic =1; which is clearly satisfied here.

The ‘Z’ or floating values have not been captured at the outputs due to the presence of the CMOS inverter. Hence no abrupt outputs are found and the system is working like a Half adder.

The advantage of using domino logic is the power it holds with precharge

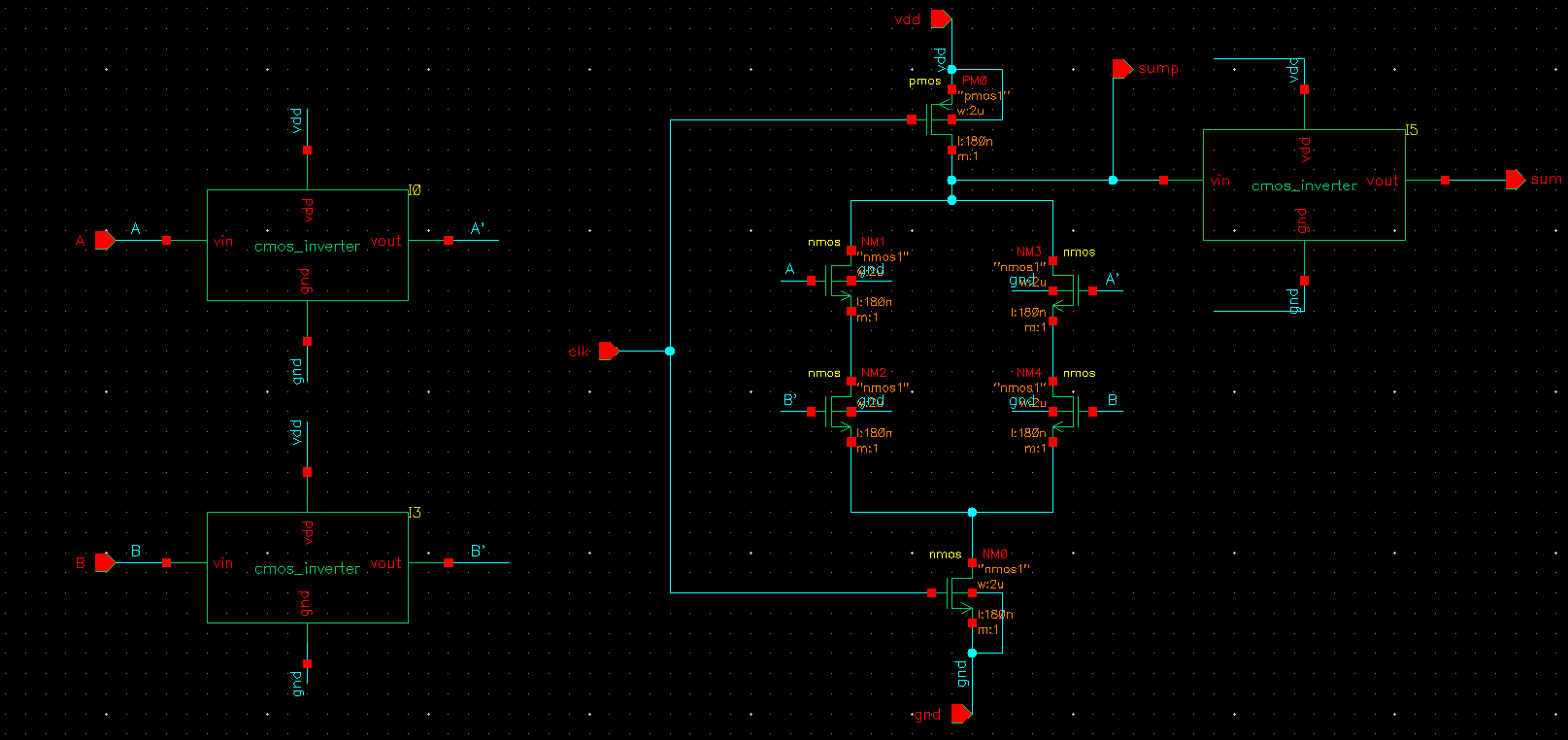
i.e., when the clk=0 the output does not go to ‘zero’ as PMOS will be on in this stage and the output will be charged to VDD.

Transient Response – with precharge output

As one can observe in the output provided above, sump refers to the output when clk=0; this output has been estimated before giving the CMOS inverter logic. i.e., before the domino logic.

The precharge phase is when the clk=0; and sump holds the value of VDD; 1.8V as per the design.

The evaluation phase begins when the clk=1;

The schematic for the sum that has calculated sump value is as follows:

Hence this simulation clearly states that the circuit will perform its job even when clock input=0 due to the precharge phase [ the presence of gated clock through which PMOS gets activated and the sump takes the value that the VDD holds.

# Conclusion

The completion of the project involves the implementation of a half adder using domino logic by which the floating values are avoided when clk=1; data=0.

The observation on failures faced in the project were errors in the gate-level implementation of the sum inputs leading to incorrect results in the overall adder and the difference of complementary pins at the schematic level led to semantic error in the design ,nevertheless all the errors were successfully rectified.

Notable success in the project were Design Implementation, Rigorous simulation testing was conducted to validate the correctness and functionality of the half adder. The testbench covered a wide range of input scenarios, ensuring that the design met the specified requirements along with conceptual understanding were some notable outcomes too.

# References

Textbook: Digital design and computer architecture [second edition] by David Money Harris & Sara L. Harris

Textbook: Sung-Mo (Steve) Kang, Yusuf Leblebigi, “CMOS

Digital Integrated Circuits Analysis and Design”

# Acknowledgement

I am deeply thankful to Mr. Chiranjeevi Sir, professor for the subject of Digital VLSI [Very Large-Scale Integration] Design , [a 4th semester course], for their guidance, encouragement, and insightful feedback. Their expertise and willingness to share their knowledge have been instrumental in shaping the direction of this project.

I would also like to express my gratitude to my peers who provided a supportive and collaborative environment. Their constructive feedback and shared insights have enhanced the overall quality of this work.

This work would not have been possible without the collective efforts and support of all mentioned above. Thank you for your valuable contributions.