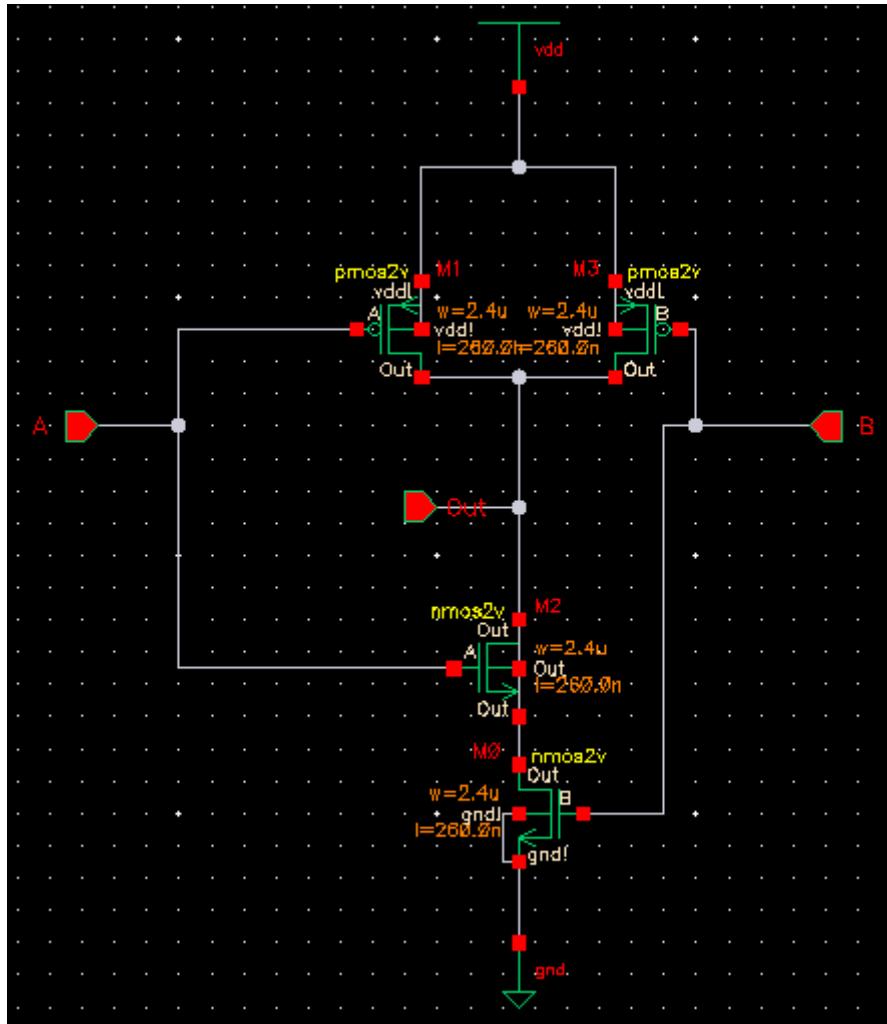


Alunos: Henrique Meurer Zardo e Lucas Tavares Rockembach
Período: 2025-1

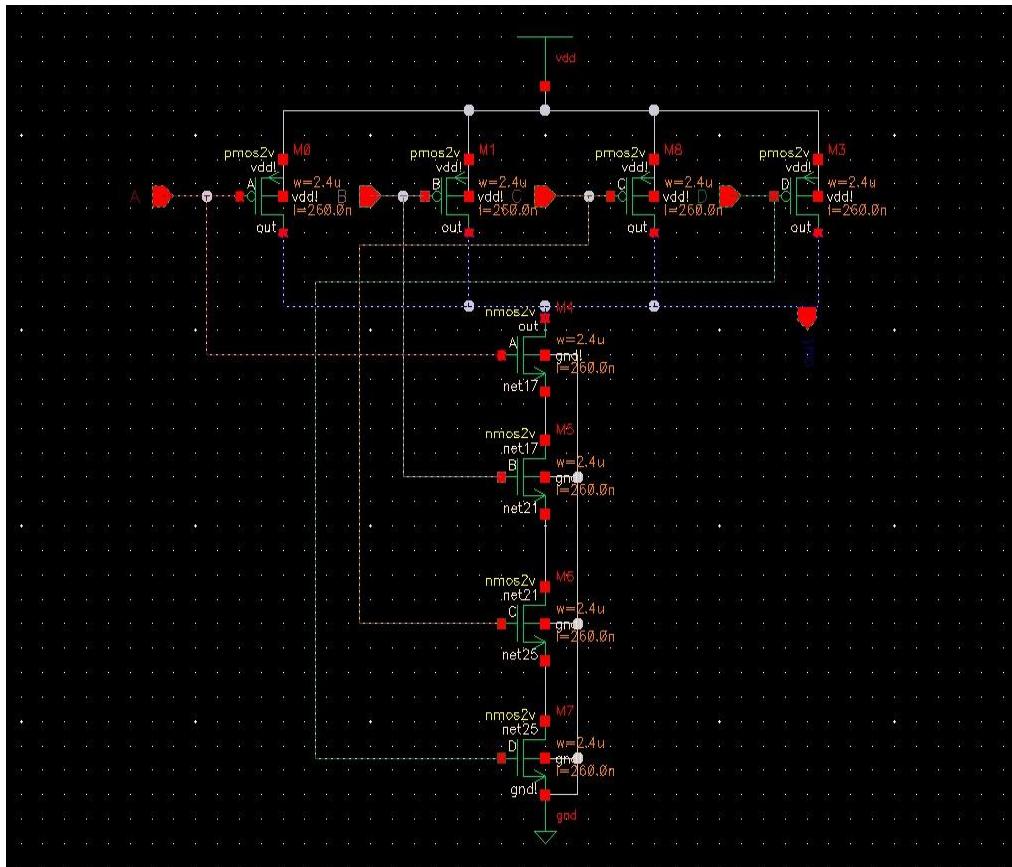
5.1) Portas Lógicas CMOS

a) NAND de 2 e 4 entradas

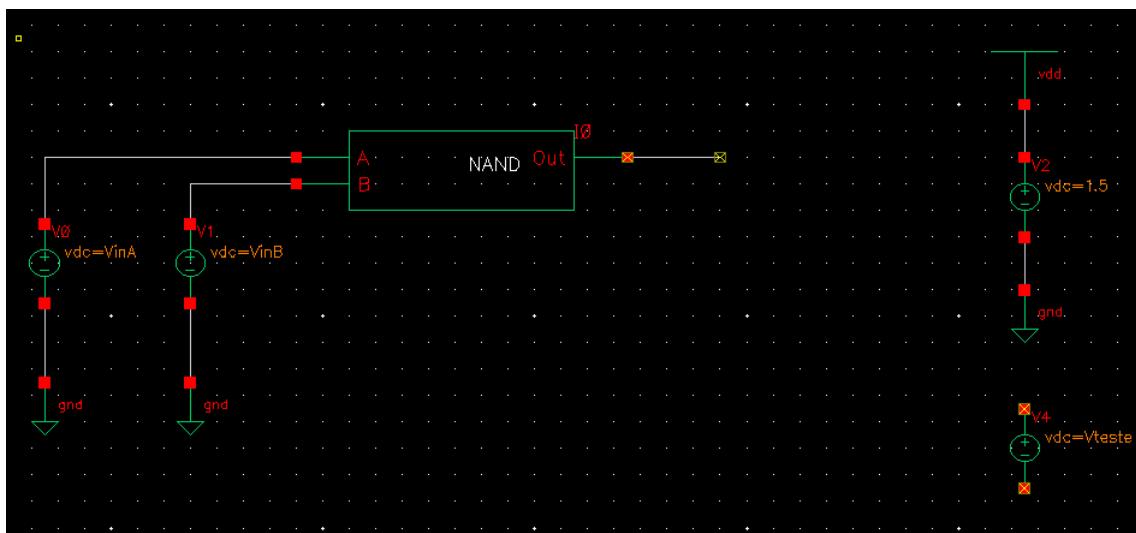
Foi criado o esquemático de uma porta lógica Nand de 2 entradas:



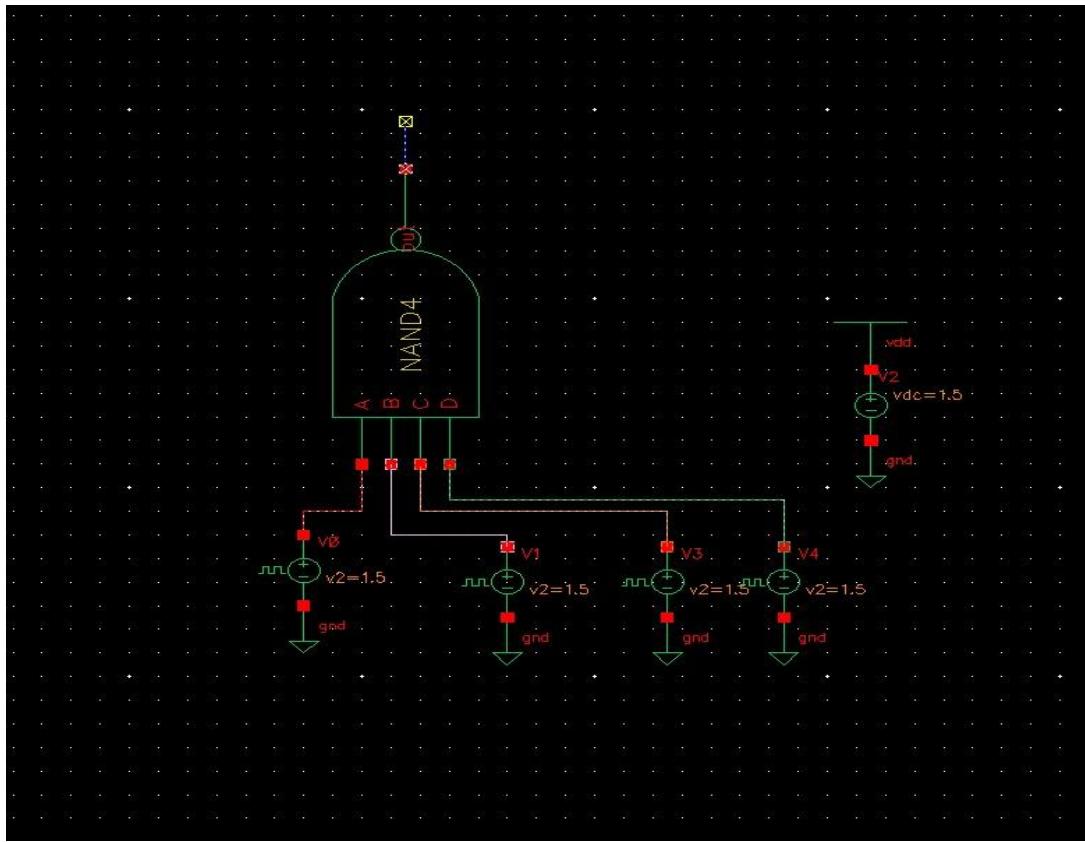
Em seguida, analogamente foi feito o mesmo para a de 4 entradas:



Dessa forma, esses foram os circuitos de teste que possibilitaram evidenciar o devido funcionamento das portas lógicas:



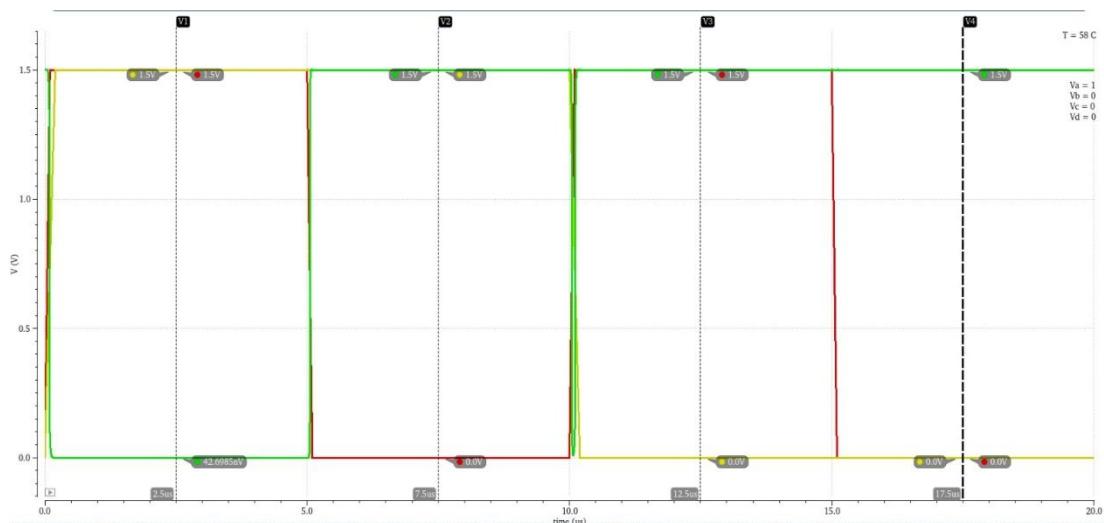
Obs: Circuito de teste para Nand de 2 entradas, o retângulo escrito NAND é o símbolo da porta lógica.



Obs: Circuito de teste para Nand de 4 entradas, a forma geométrica com NAND4 escrita, é o símbolo da porta lógica.

Então, assim ficaram as simulações do funcionamento para ambas:

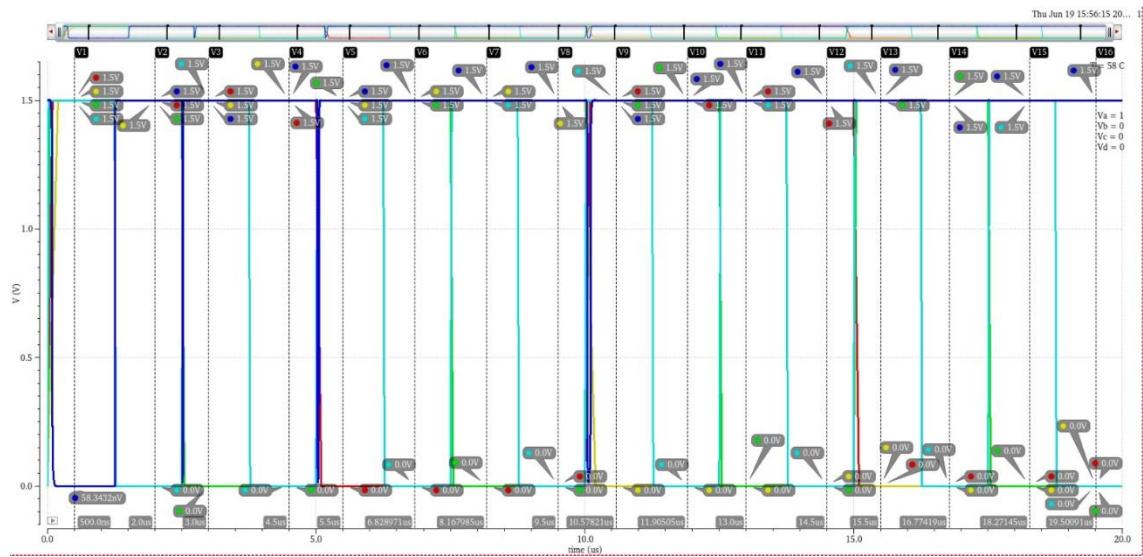
Nand de 2 entradas:



Verde: Saída, vermelho: entrada A, amarelo: entrada B.

Assim, é possível perceber que a saída assume nível lógico baixo apenas quando ambas as entradas estão em nível lógico alto (1,5-1,5), permanecendo em nível alto para as demais combinações (0-0; 0-1,5; 1,5-0).

Nand de 4 entradas:



Azul-escuro: saída, azul-claro: entrada A, verde: entrada B, vermelho: entrada C, amarelo: entrada D.

Assim, é possível perceber que a saída assume nível lógico baixo apenas quando as quatro entradas estão em nível lógico alto (1,5-1,5-1,5-1,5), permanecendo em nível alto para as demais combinações.

b e c) AOI21

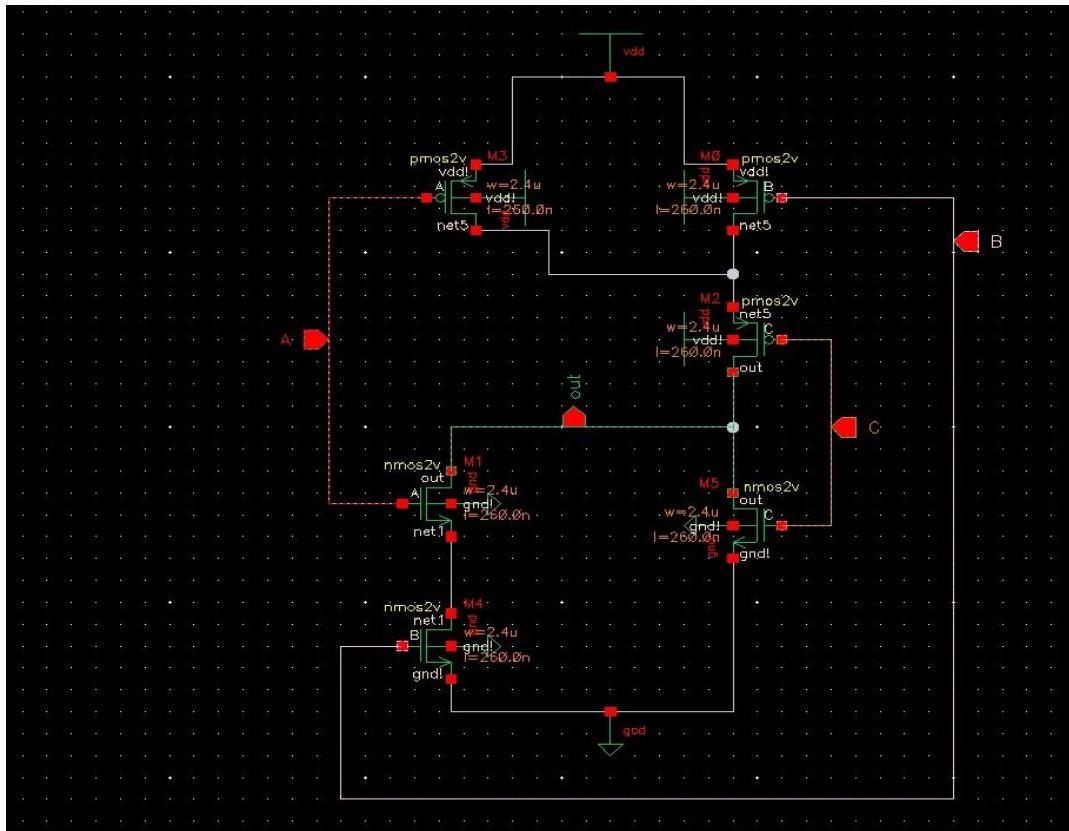
Sabe-se que a porta **AOI21** (AND-OR-Invert 2:1) implementa a função:

$$Y = \overline{(A \cdot B)} + C$$

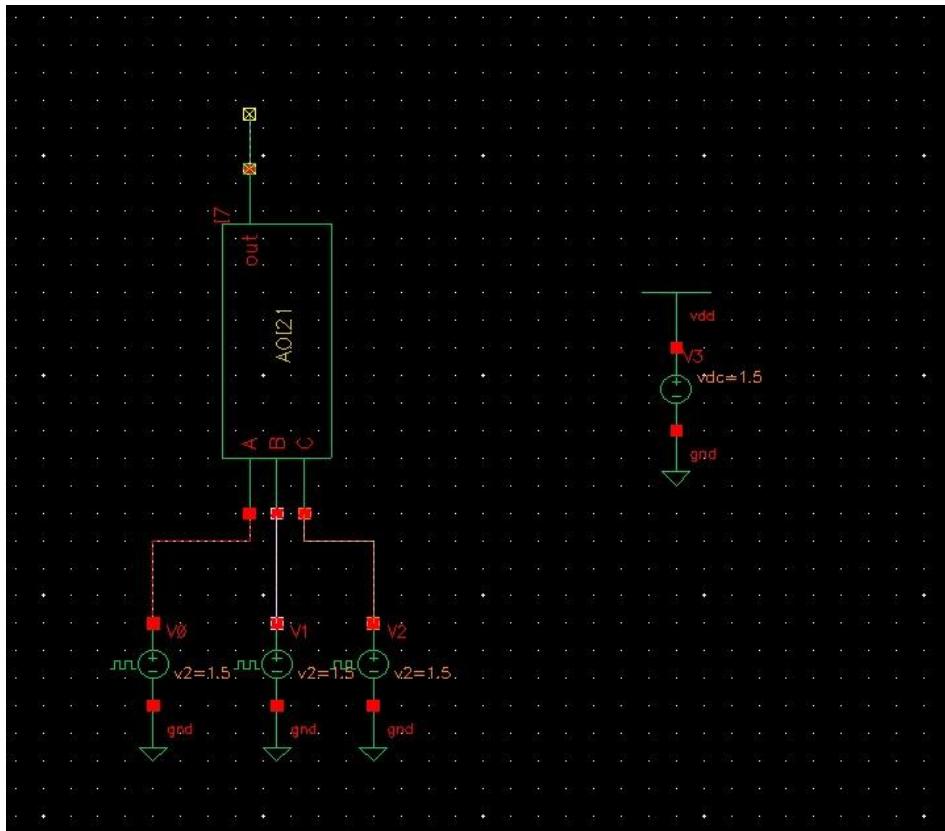
Assim, pelos postulados da álgebra booleana, a tabela verdade da expressão torna-se:

Entrada A	Entrada B	Entrada C	Saída Y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Por isso, assim ficaram o esquemático:

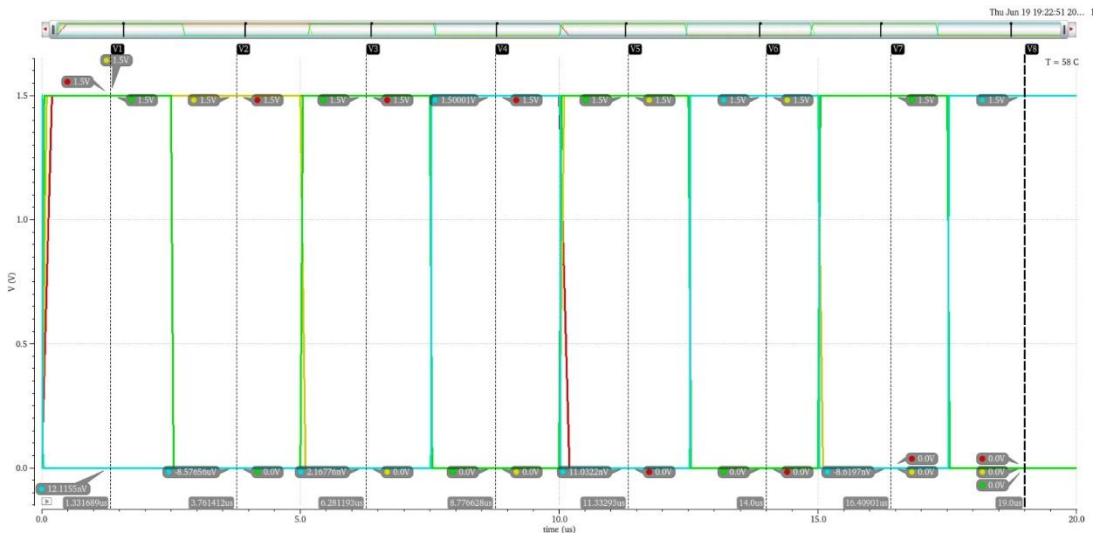


E o circuito de teste:



Obs: Circuito de teste para AOI21, o retângulo escrito AOI21 é o símbolo da porta lógica.

A simulação obteve os seguintes resultados gráficos:



azul-claro: saída, **vermelho:** entrada A, **amarelo:** entrada B, **verde:** entrada C.

Dessa forma, é possível notar o comportamento adequado da porta lógica de acordo com a tabela verdade: ela assume nível lógico alto quando todas as entradas estão desligadas (0-0-0), ou quando somente a B está ligada (0-1,5-0), ou quando somente a A está ligada (1,5-0-0). Para todos os demais casos, a saída permanece em nível lógico baixo, pois A & B estariam ligadas juntas ou C estaria ligada.

d) AOI22

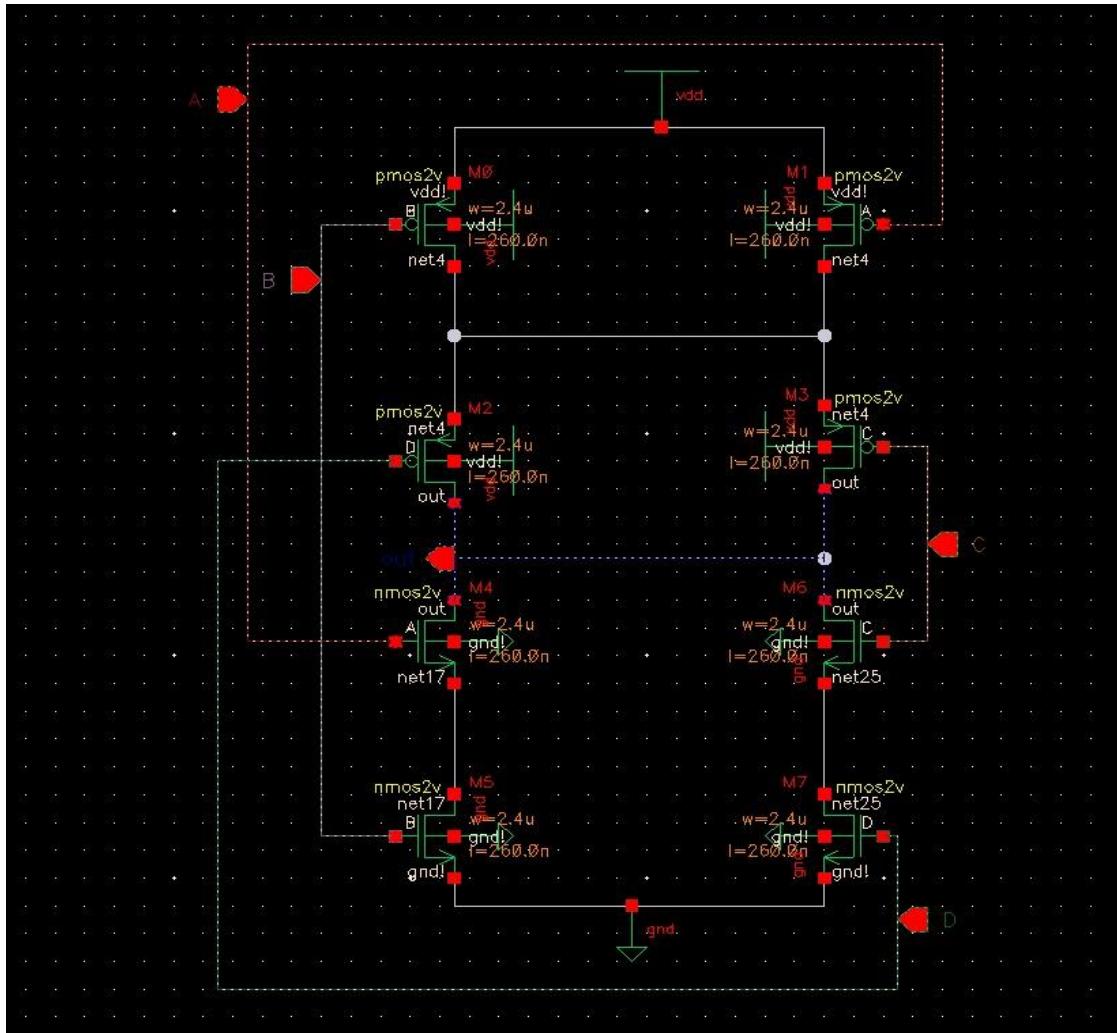
Sabe-se que a porta **AOI22** (AND-OR-Invert 2:2) implementa a função:

$$Y = \overline{(A \cdot B) + (C \cdot D)}$$

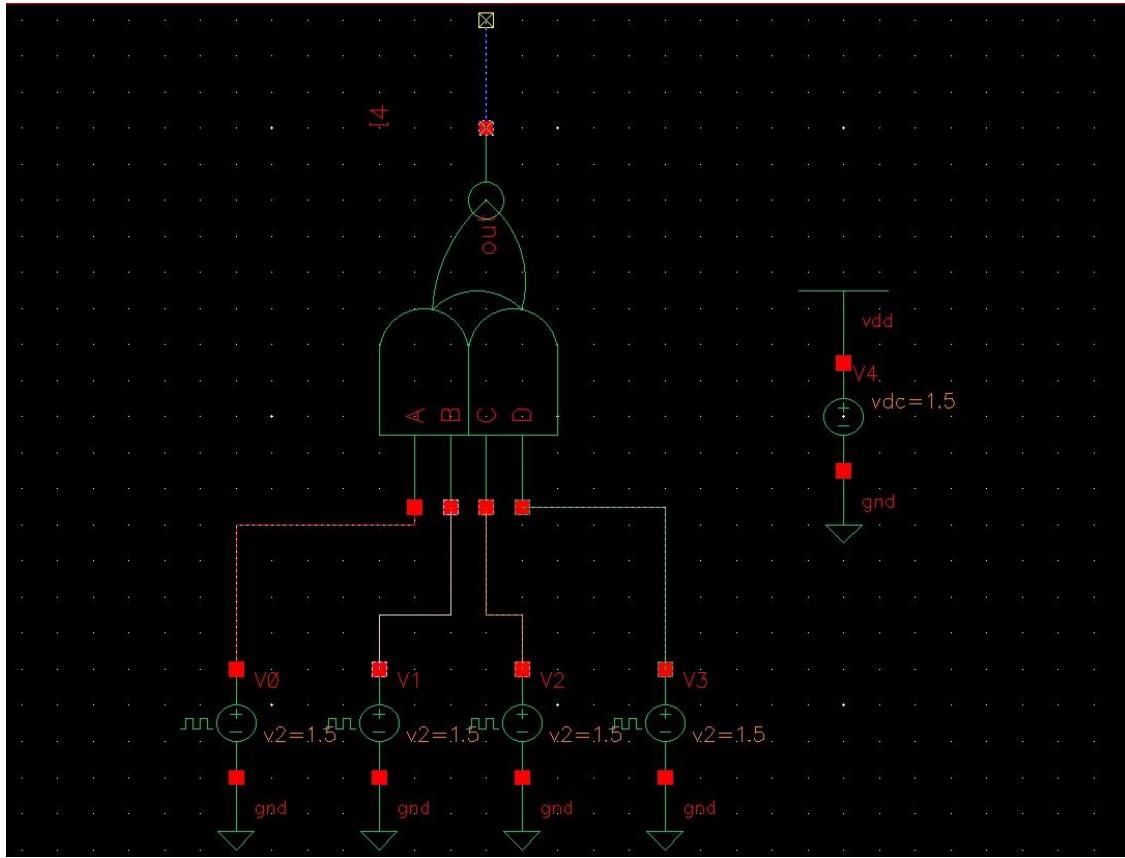
Desse modo, pelos postulados da álgebra booleana, a tabela verdade da expressão é:

Entrada A	Entrada B	Entrada C	Entrada D	Saída Y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Por isso, assim ficaram o esquemático:

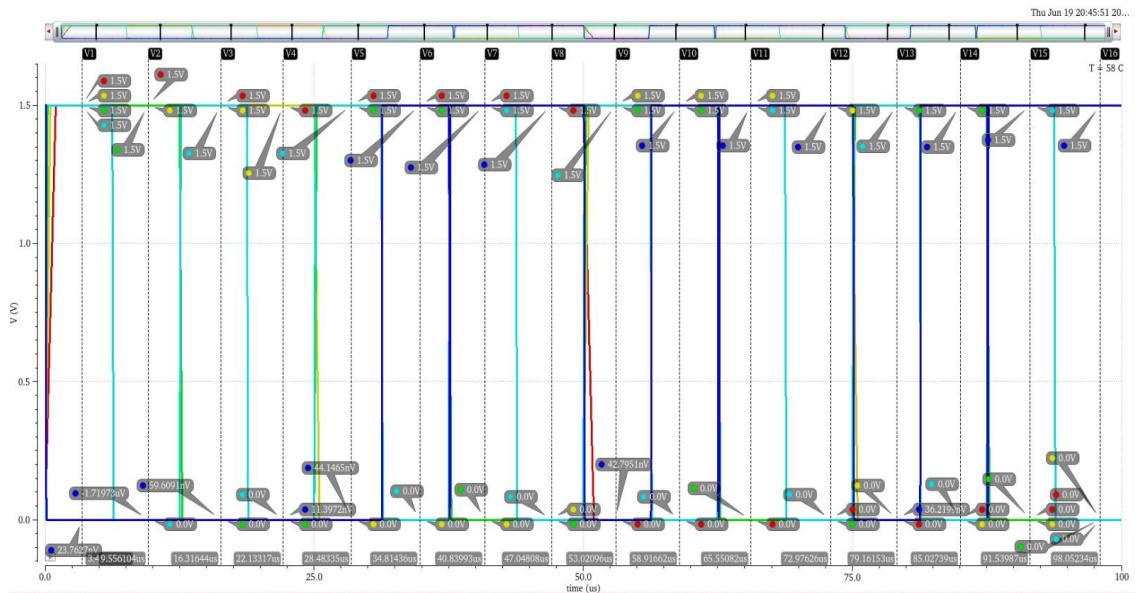


E o circuito de teste:



Obs: Circuito de teste para AOI22, a forma com 2 portas AND, uma OR e uma NOT é o símbolo da porta lógica.

Estes foram os resultados obtidos pela simulação:



Azul-escuro: saída, vermelho: entrada A, amarelo: entrada B, verde: entrada C, azul-claro: entrada D.

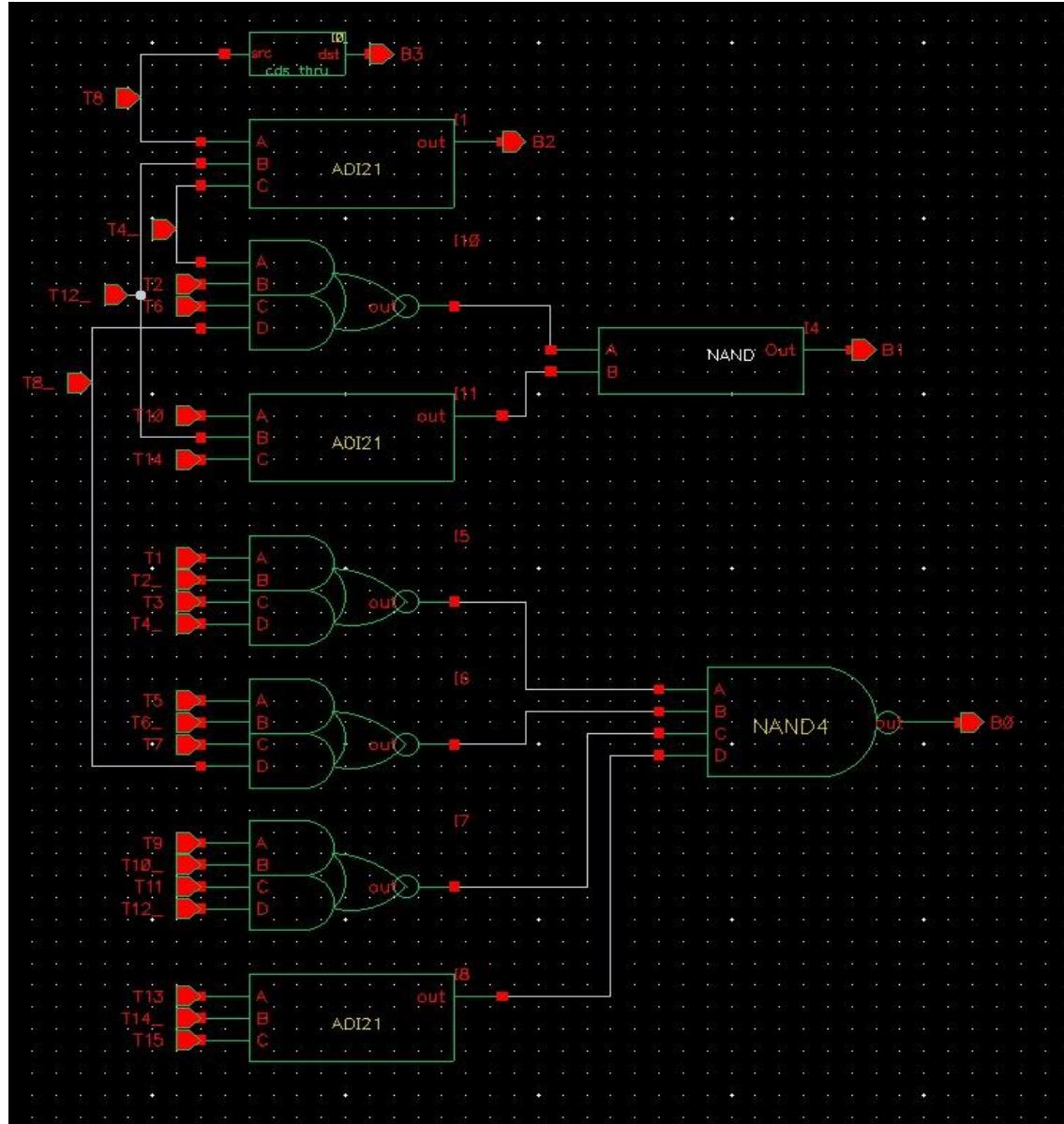
Portanto, é perceptível o funcionamento adequado da porta lógica, assumindo nível lógico alto para as combinações (0-0-0-0), (0-0-0-1,5), (0-0-1,5-0), (0-1,5-0-0), (0-1,5-0-1,5), (0-1,5-1,5-0), (1,5-0-0-0), (1,5-0-0-1,5) e (1,5-0-1,5-0), ou seja, para todas em que A &

B não são ligadas simultaneamente nem que C & D são ligadas simultaneamente. Assim, para as demais combinações, a saída permanece em nível lógico baixo, pois elas implicam ou em AB ligadas juntas ou em CD ligadas juntas.

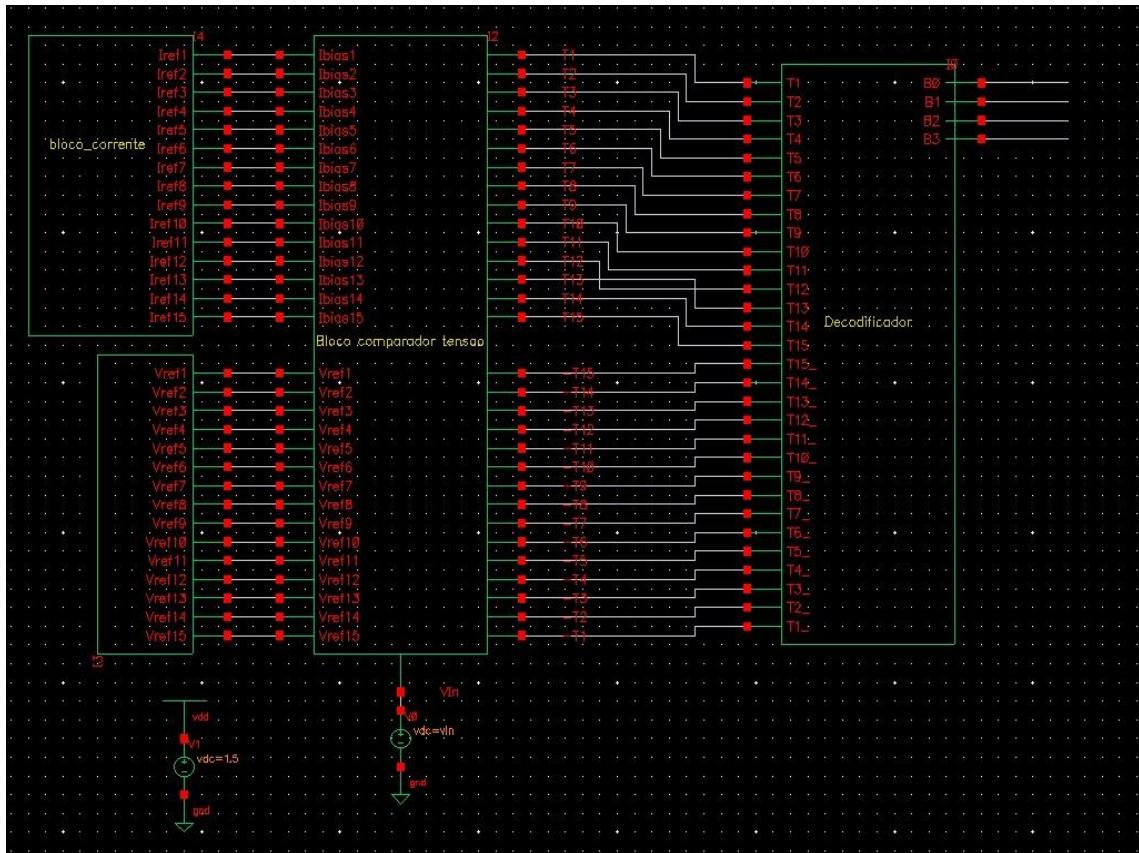
5.2) Decodificador

a) Esquemático e símbolo

Assim ficou o esquemático do decodificador de código termométrico para binário, usando as portas projetadas:



E o circuito de teste foi confeccionado dessa maneira:

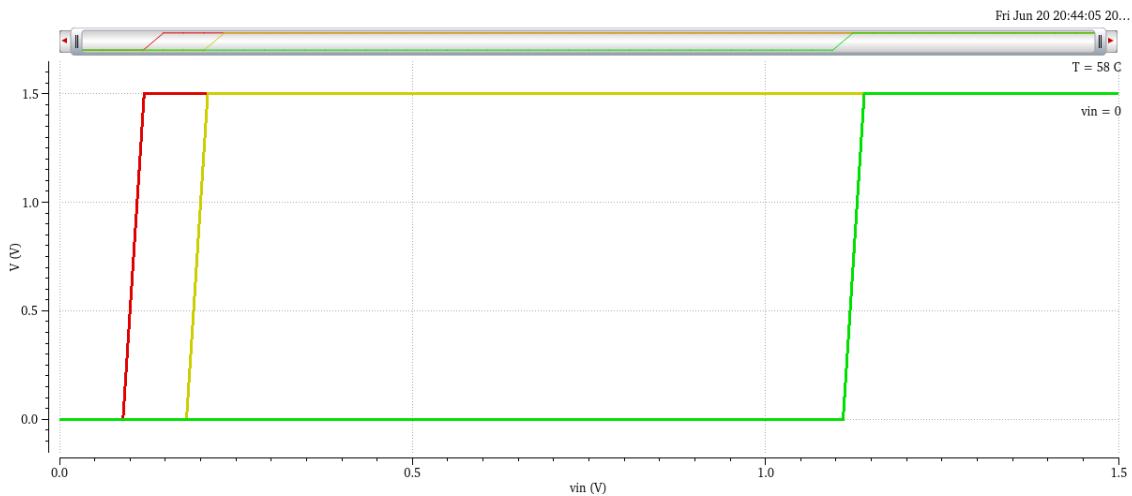


Obs: O bloco escrito “Decodificador” é o símbolo criado para o decodificador.

b) Descrição do código e simulação

Primeiramente, sabe-se que o código termométrico (ou unário), é uma forma de representação binária onde a quantidade de bits em nível lógico alto indica diretamente a magnitude de um valor. No contexto específico do decodificador do relatório, as entradas T1 a T15 correspondem às saídas de um comparador em um ADC, em que T1 representa o menor nível de tensão comparado e T15 o maior. Assim, o padrão válido do código termométrico consiste em uma sequência de bits ligados a VDD, iniciando em T1 e se estendendo de forma contínua até uma posição Tn, seguida apenas por zeros até T15. Ou seja, não há transições de 0 para VDD dentro da sequência: uma vez que um zero aparece, todos os bits seguintes devem permanecer em zero. Portanto, esse método é útil em conversores analógico-digitais porque reflete diretamente quantos níveis de tensão foram superados, facilitando a conversão posterior para o código binário equivalente.

A simulação do circuito apresentou as seguintes curvas:



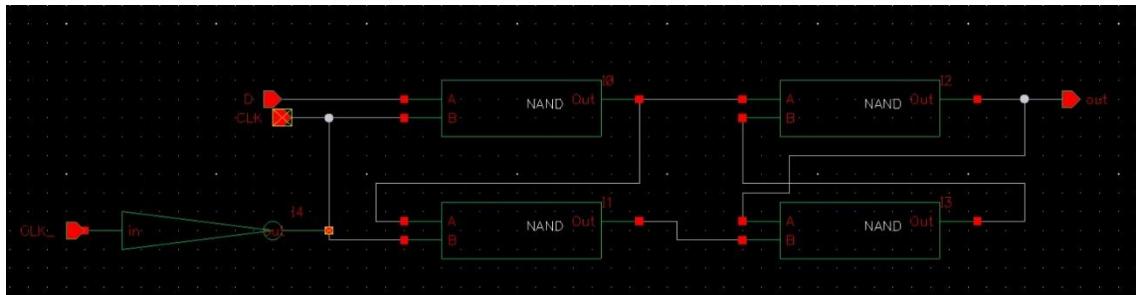
Vermelho: B0, amarelo: B1, verde: B2.

Ao analisar o gráfico, nota-se que cada divisória corresponde a um dos possíveis estados das combinações de saída. Assim, verifica-se que o decodificador opera corretamente e, em conjunto com o comparador, demonstra ser capaz de realizar a conversão de sinais analógicos para digitais.

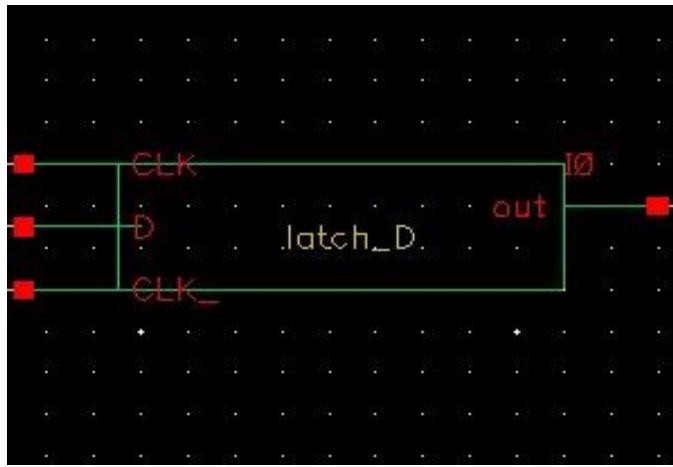
5.3) Flip-flop e Registrador

a) Esquemáticos

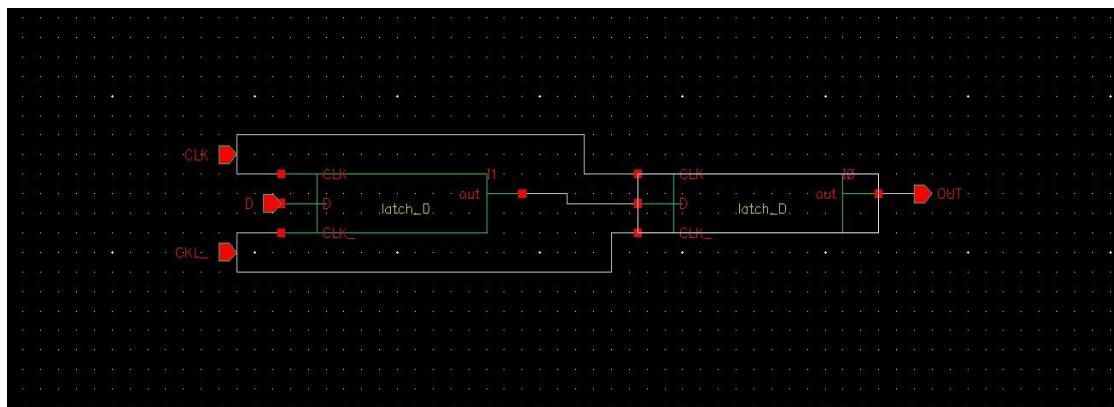
Segue abaixo o esquemático do latch utilizando as portas lógicas criadas:



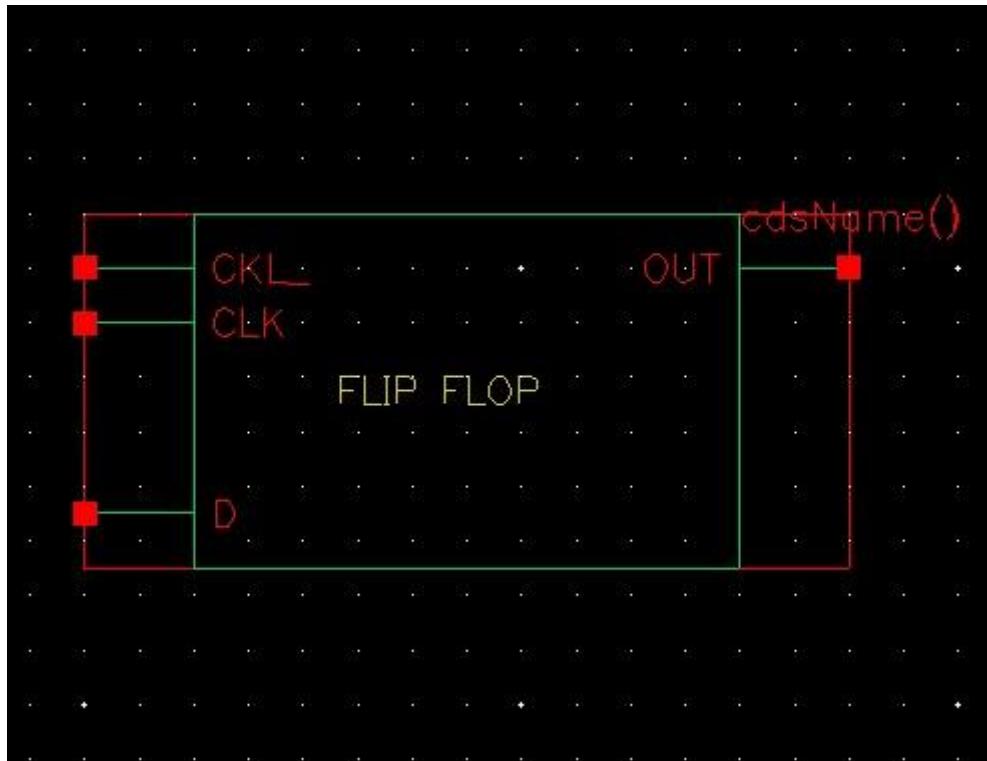
Em seguida, foi criado o seu símbolo correspondente:



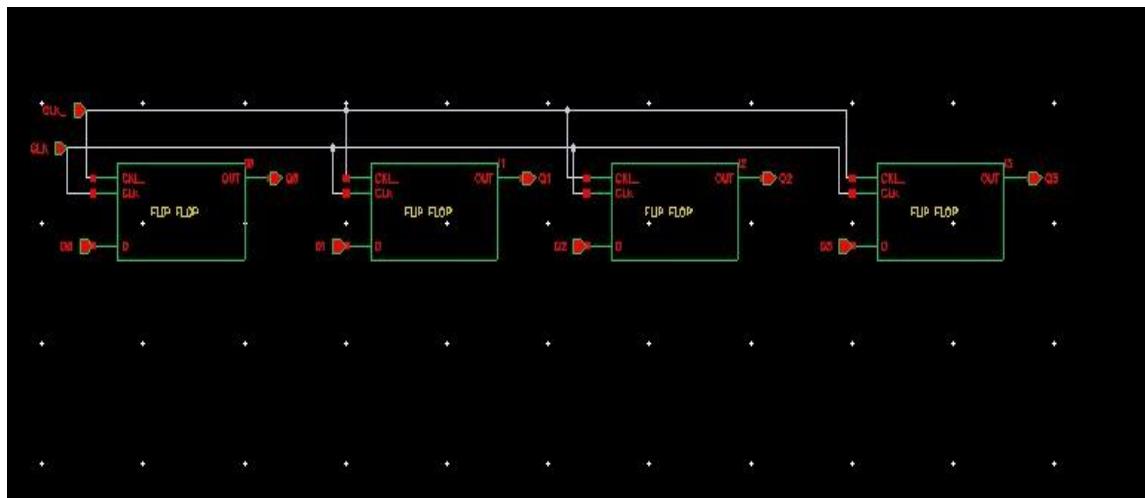
Após isso, foi projetado o esquemático para o flip-flop:



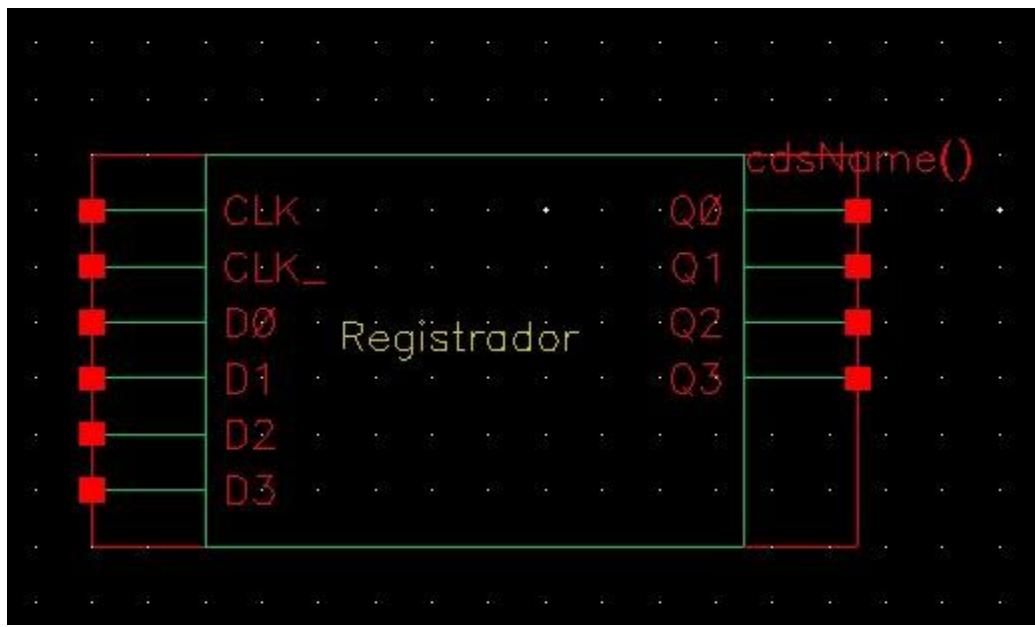
Com seu respectivo símbolo:



Por fim, utilizando os flip-flops foi projetado o registrador de 4 bits:

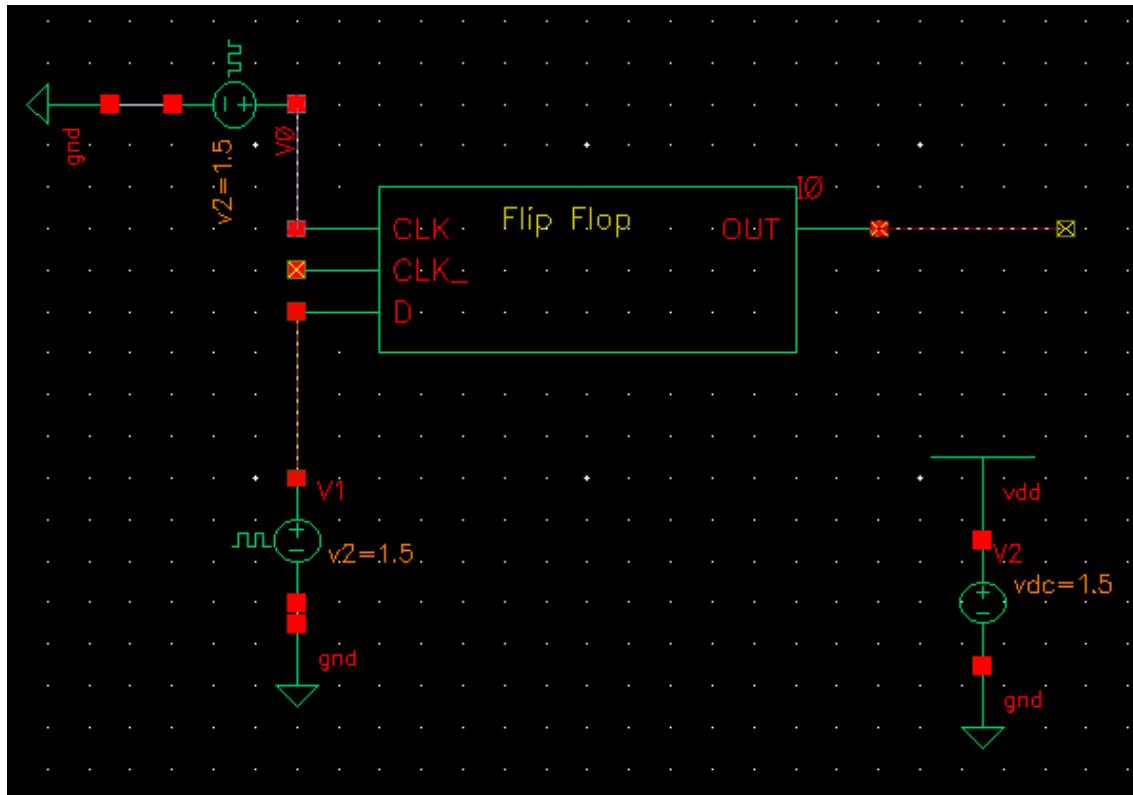


Cujo símbolo ficou da seguinte forma:



b) Simulação

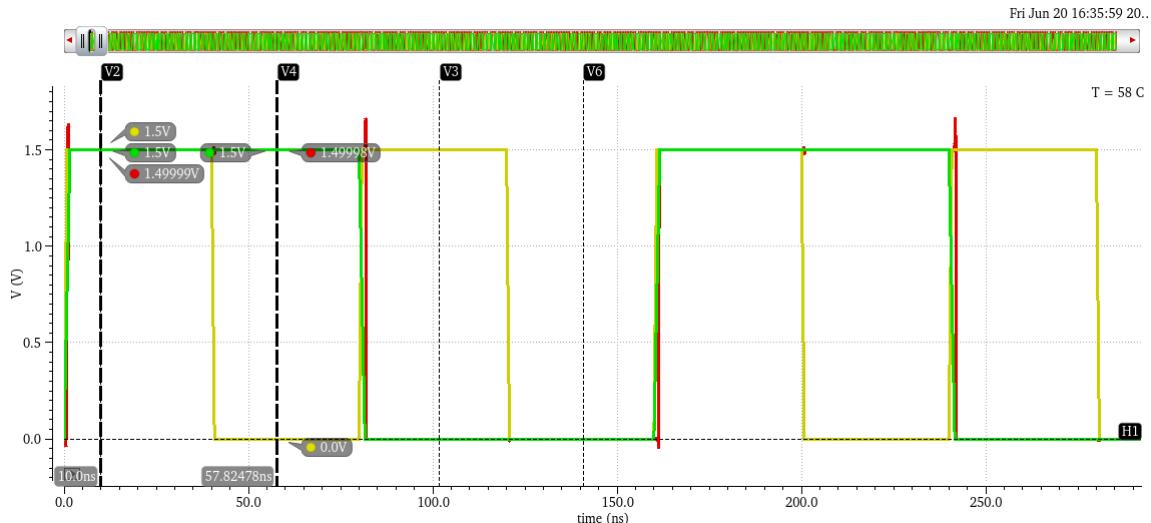
Esse foi o circuito de teste para o flip-flop:

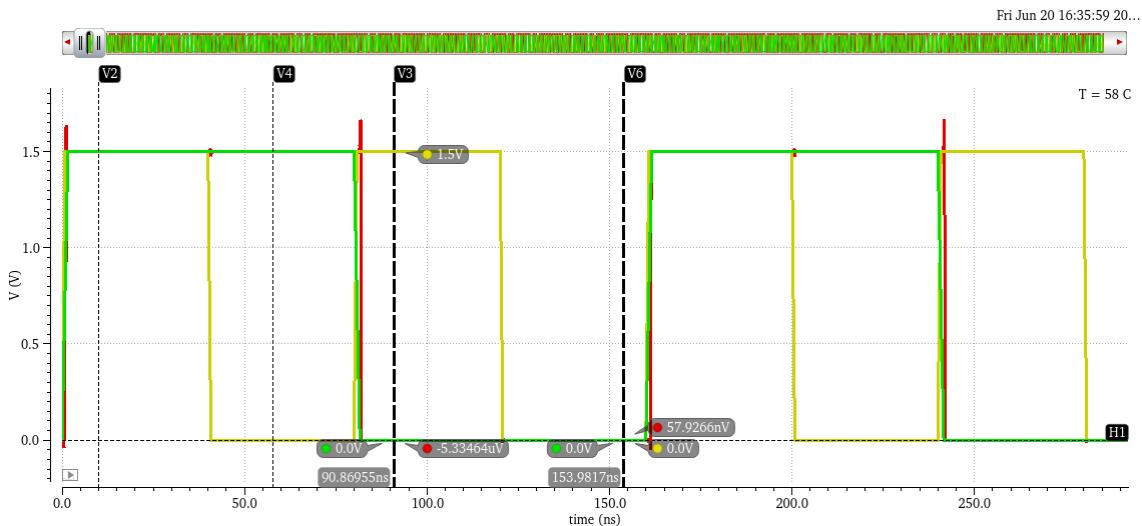


Com um período de 80 ns para a fonte aplicada ao clock, o que corresponde à determinação de 12,5 MHz de frequência de clock:

CDF Parameter	Value	Display
Frequency name for 1/period		off ▾
Noise file name		off ▾
Number of noise/freq pairs	0	off ▾
DC voltage		off ▾
AC magnitude		off ▾
AC phase		off ▾
XF magnitude		off ▾
PAC magnitude		off ▾
PAC phase		off ▾
Voltage 1	0 V	off ▾
Voltage 2	1.5 V	off ▾
Period	80n s	off ▾
Delay time		off ▾
Rise time		off ▾
Fall time		off ▾
Pulse width		off ▾
Temperature coefficient 1		off ▾

Então, seguem nas duas imagens abaixo os resultados da simulação para as diferentes combinações de níveis lógicos:





Vermelho: saída, verde: pino D, amarelo: pino clock.

Dessa forma, é possível evidenciar o funcionamento correto do flip-flop D, pois quando o clock sobe para o nível lógico alto, a saída assume o valor atual do pino D, como mostram, mais especificamente, os marcadores V2 e V3.

Conclusão:

Portanto, o desenvolvimento dos circuitos digitais propostos nesta etapa permitiu não apenas a aplicação prática dos conceitos de álgebra booleana e lógica CMOS, como também a integração desses blocos em estruturas maiores e necessárias para o funcionamento real do ADC, como decodificadores e registradores. Além disso, as simulações comprovaram o funcionamento correto de todas as portas lógicas projetadas, além da fidelidade do decodificador termométrico-binário ao representar adequadamente os estados do ADC. Por fim, o registrador de 4 bits, construído a partir de flip-flops D, demonstrou sua capacidade de armazenar os valores convertidos, operando de forma síncrona com o sinal de clock de 12,5 MHz. Dessa forma, todos os objetivos do projeto foram atingidos com êxito.