

# DICD 2025\_1

## 5 주차 실습 보고서

학번: 12191529

이름: 장준영

## 1. 이론

### #1. NAND

NAND(NOT AND) 게이트는 AND 게이트의 출력에 NOT 연산을 취한 형태로, 입력값이 모두 1 일 때만 출력이 0, 그 외에는 모두 출력이 1 이 되는 논리 회로다.

$A$	$B$	$A \cdot B$	$\overline{A \cdot B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

- **Pull-up Network (PMOS):** 두 개의 PMOS 가 병렬로 구성되어 있어 입력 중 하나라도 0 이면 출력이 VDD 에 연결된다.
- **Pull-down Network (NMOS):** 두 개의 NMOS 가 직렬로 연결되어 있어 두 입력이 모두 1 일 때만 출력이 GND 로 연결된다.

### #2. NOR

NOR(NOT OR) 게이트는 OR 게이트의 출력에 NOT 연산을 취한 형태로, 입력값이 모두 0 일 때만 출력이 1, 그 외에는 모두 출력이 0 이 되는 논리 회로다.

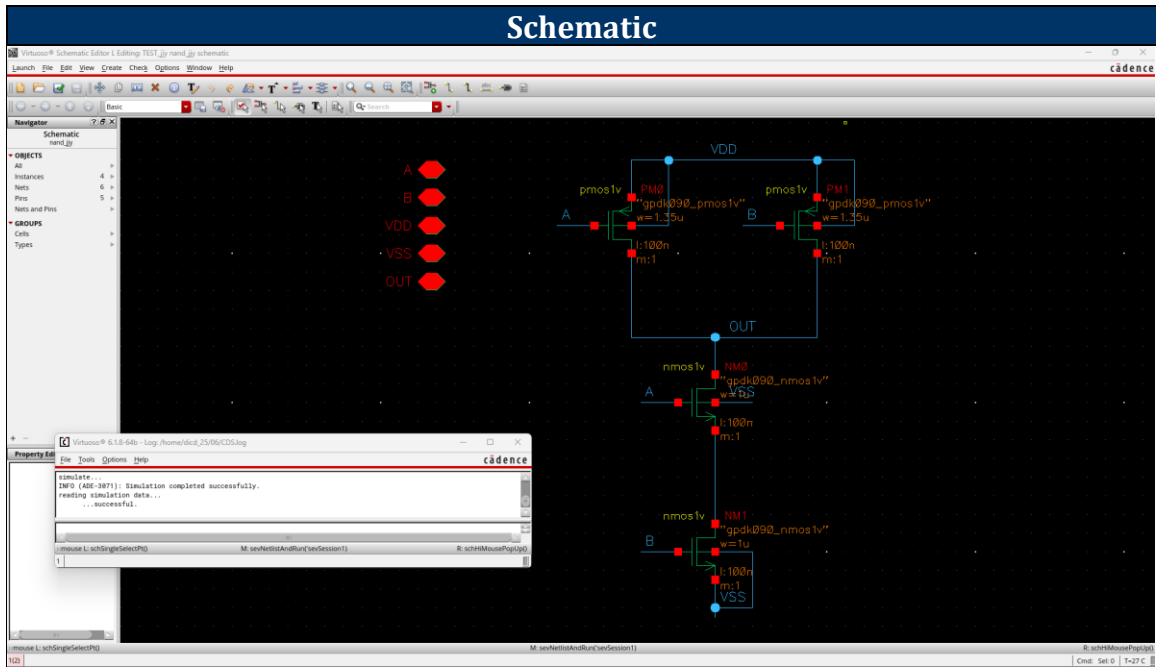
$A$	$B$	$A + B$	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

- **Pull-up Network (PMOS):** 두 개의 PMOS 가 직렬로 연결되어 있어 두 입력이 모두 0 일 때만 VDD 가 출력에 연결됨.
- **Pull-down Network (NMOS):** 두 개의 NMOS 가 병렬로 연결되어 입력 중 하나라도 1 이면 출력이 GND 에 연결됨.

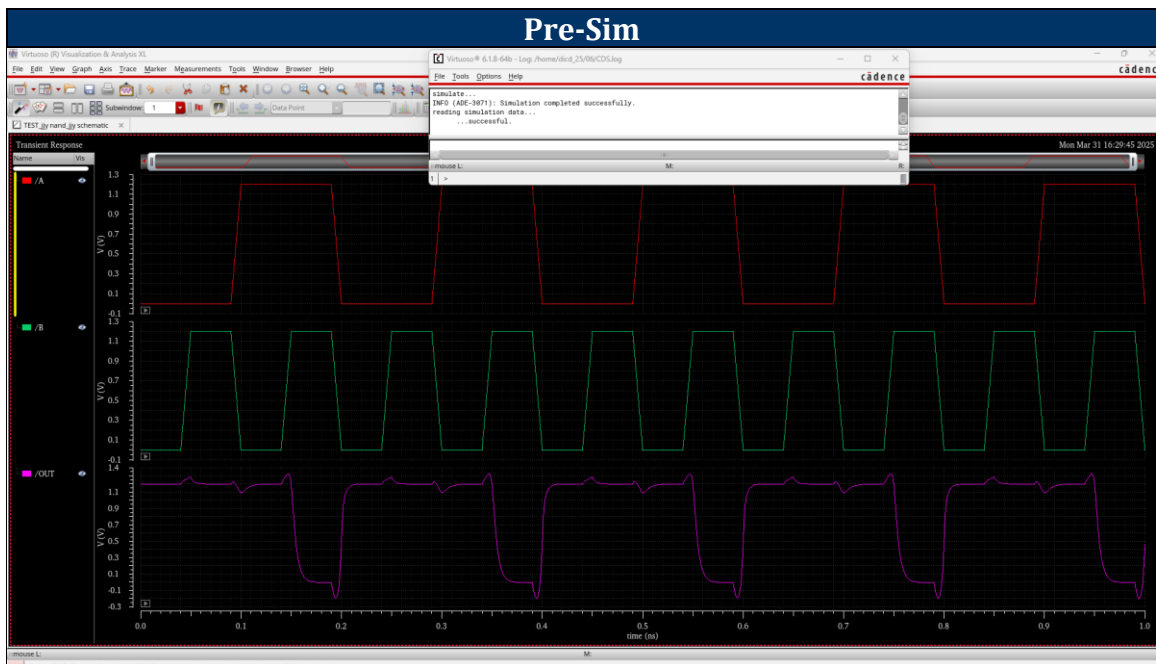
## 2. 본문

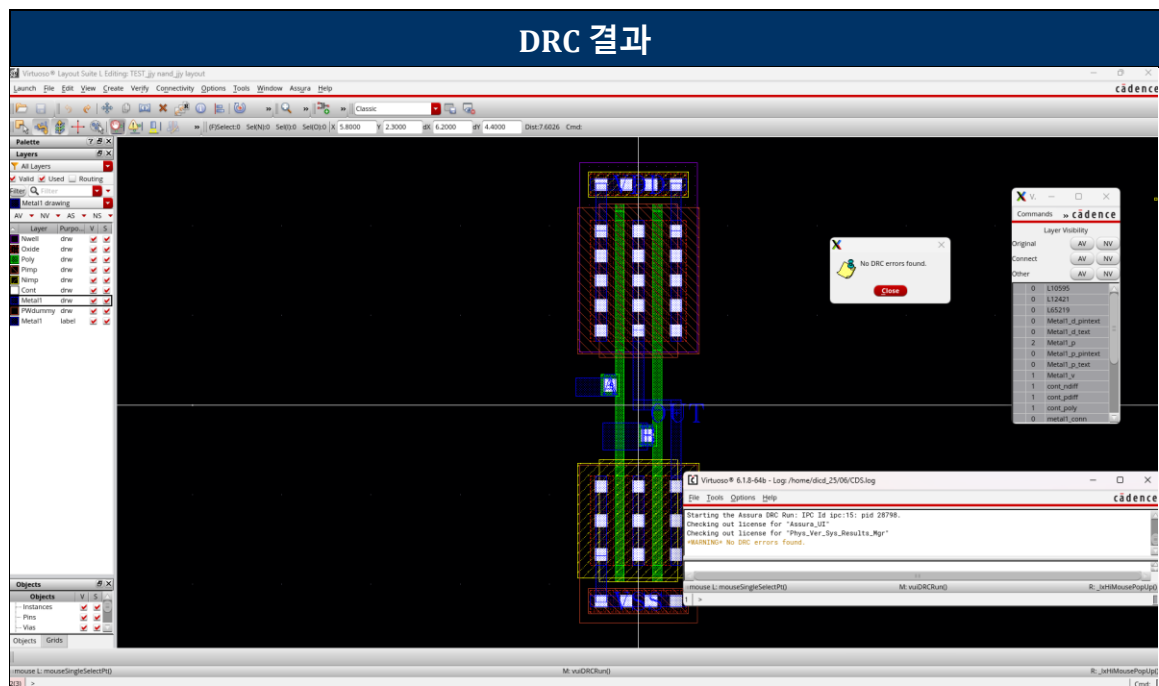
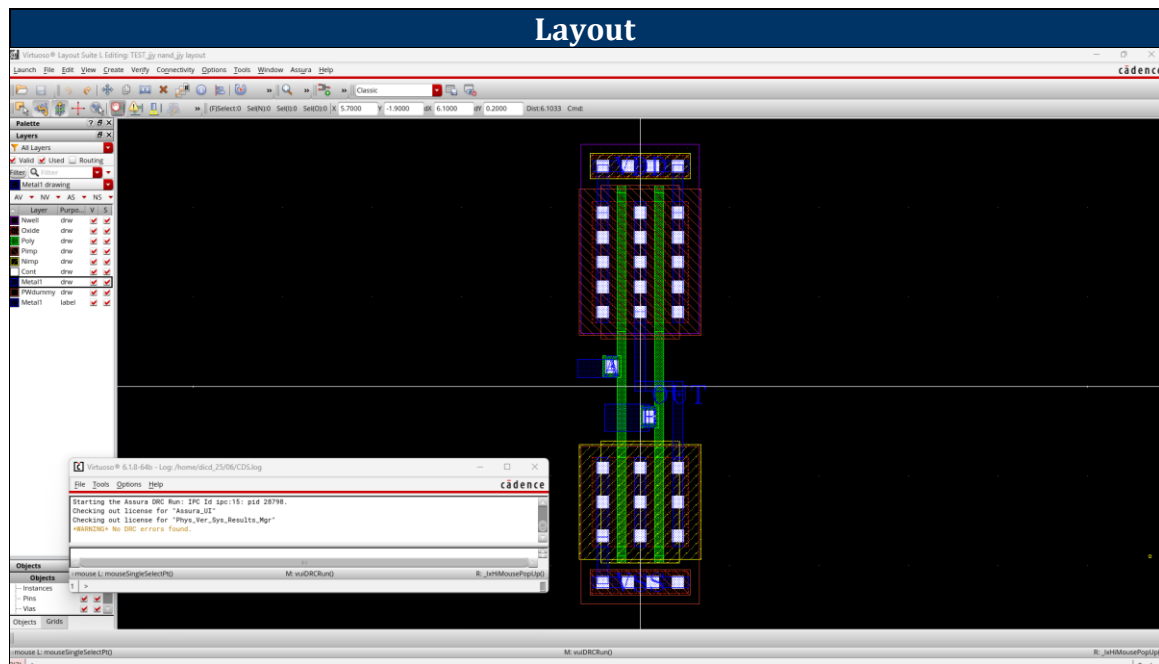
### 2-1) 실습 1: NAND

#### Schematic

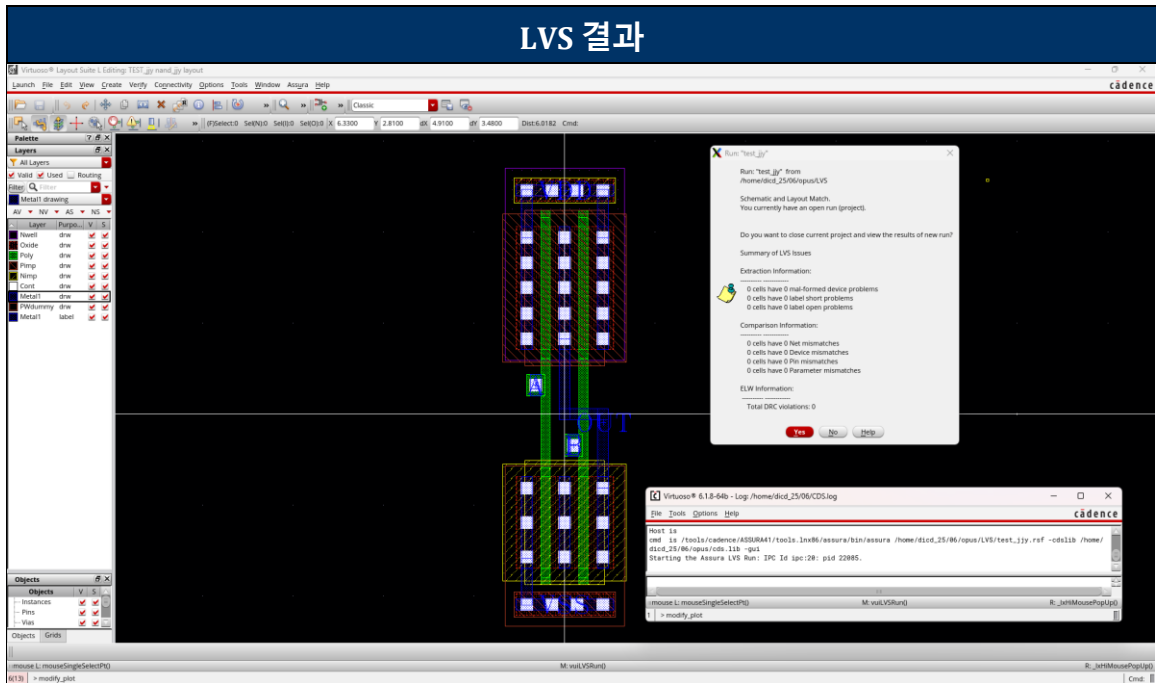


#### Pre-Sim 결과(NN)





## LVS 결과



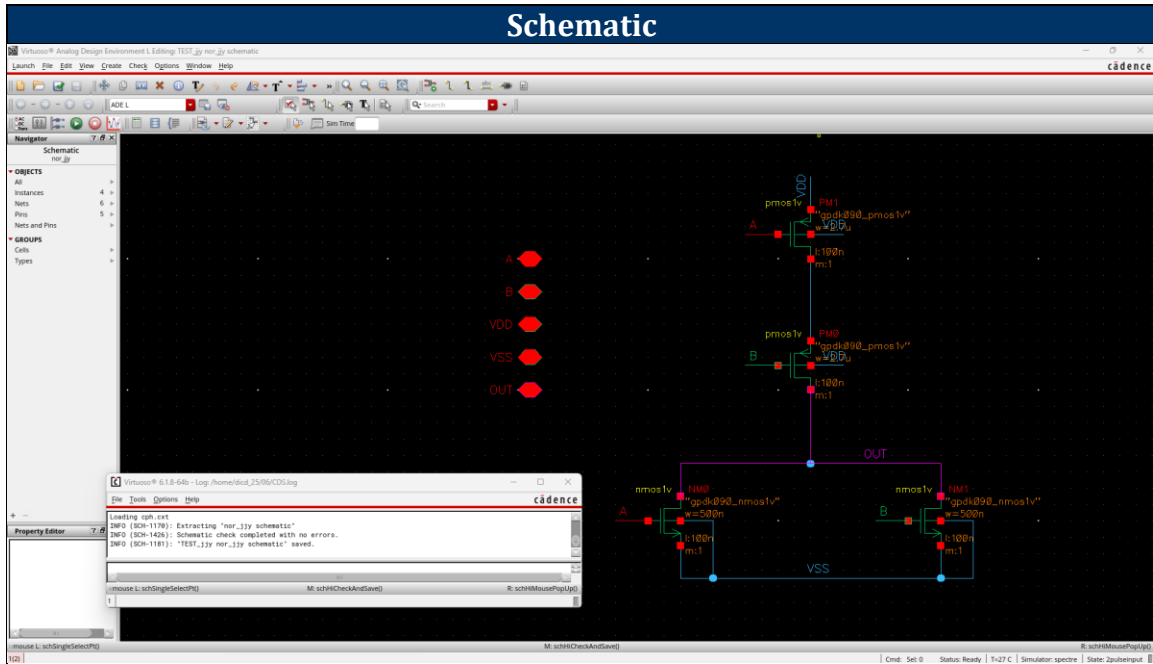
## Post-Sim 결과

### Post-Sim 결과

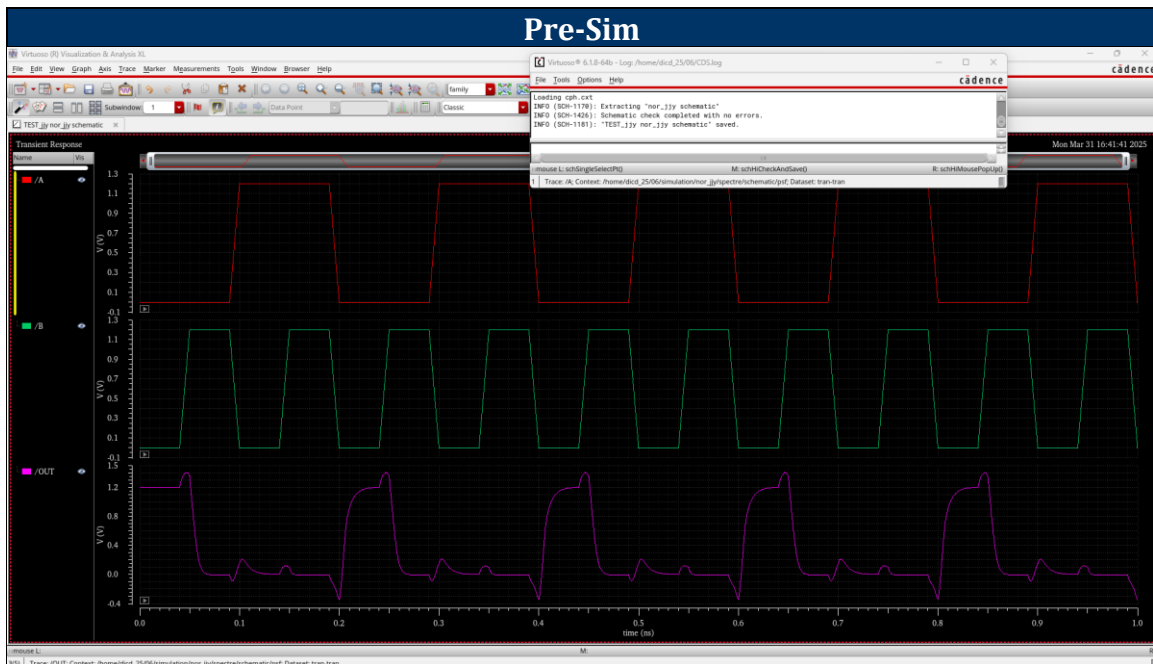
진행 안함.

## 2-2) 실습 2: NOR

### Schematic



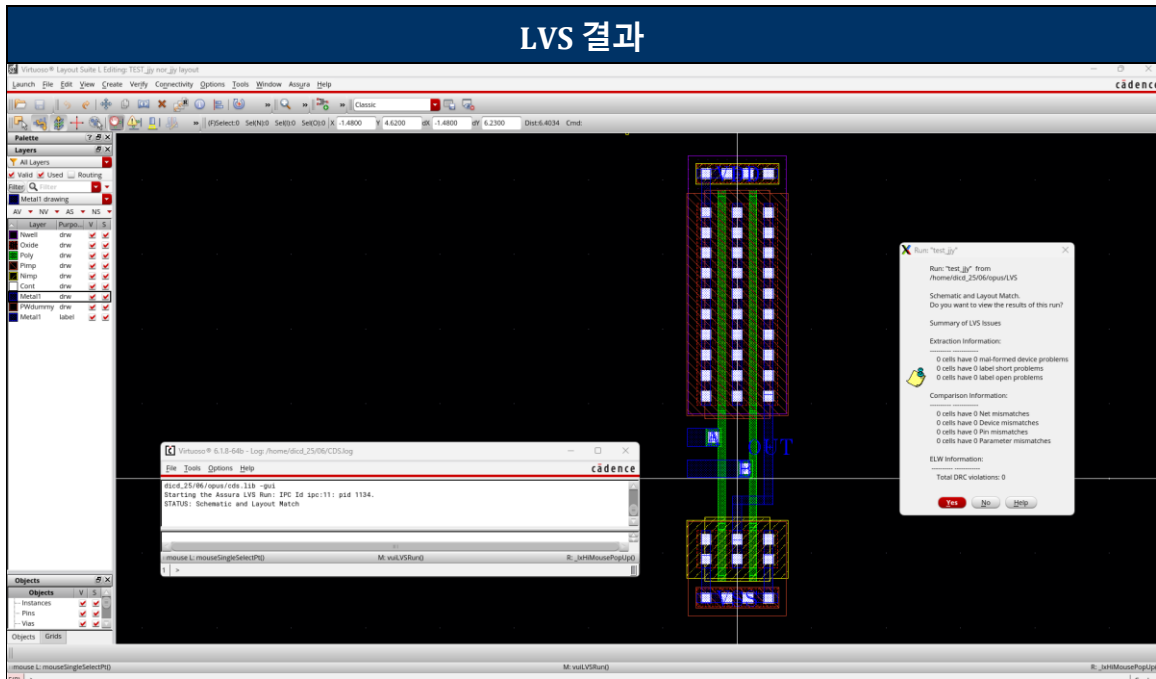
### Pre-Sim 결과(NN)



The screenshot displays the Cadence Virtuoso layout editor interface. The main workspace shows a top-down view of a rectangular PCB layout with a dense grid of components. A smaller window titled "Virtuoso 6.1.8-64b - Log: /home/dad\_25/06/CD5.log" is open, showing the startup sequence and license check for the Assura DRC tool. The status bar at the bottom indicates the current layer is "Metal1" and the design is "M\_vu0RCRun0".

# DRC 결과

## LVS 결과



## Post-Sim 결과

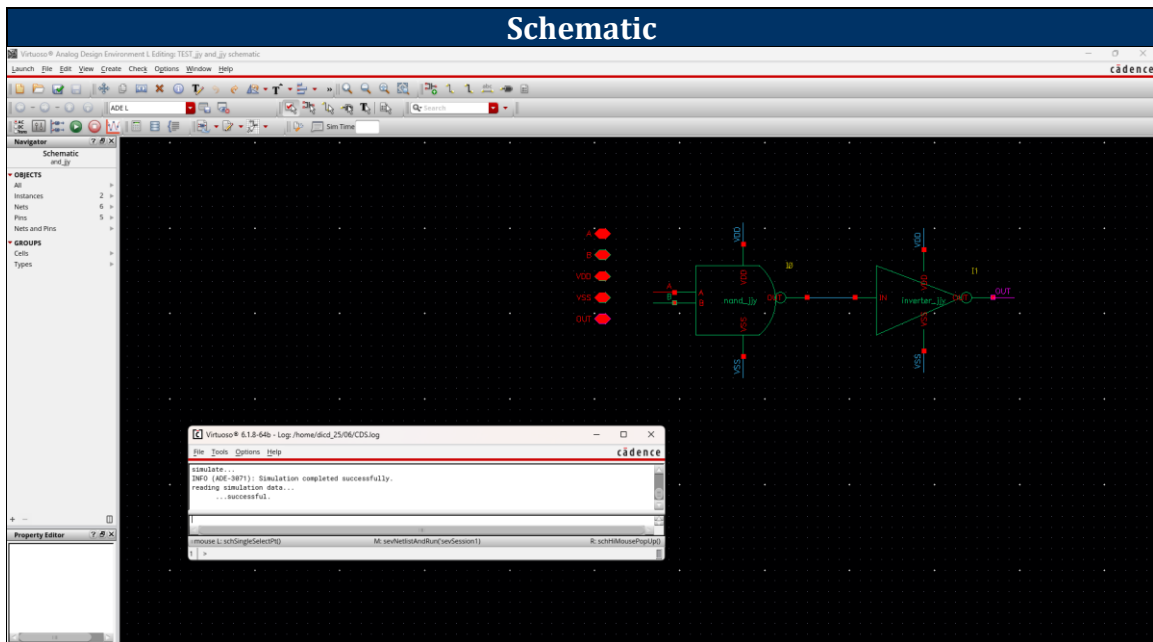
### Post-Sim 결과

진행 안함.

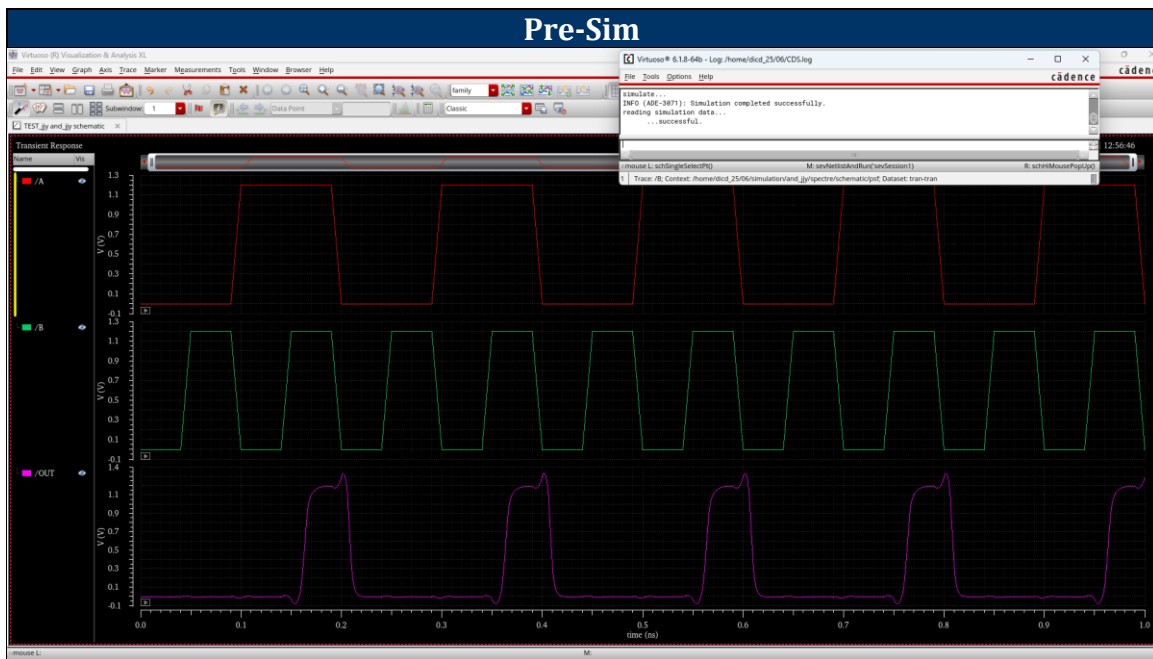


## 2-3) 실습 3: AND

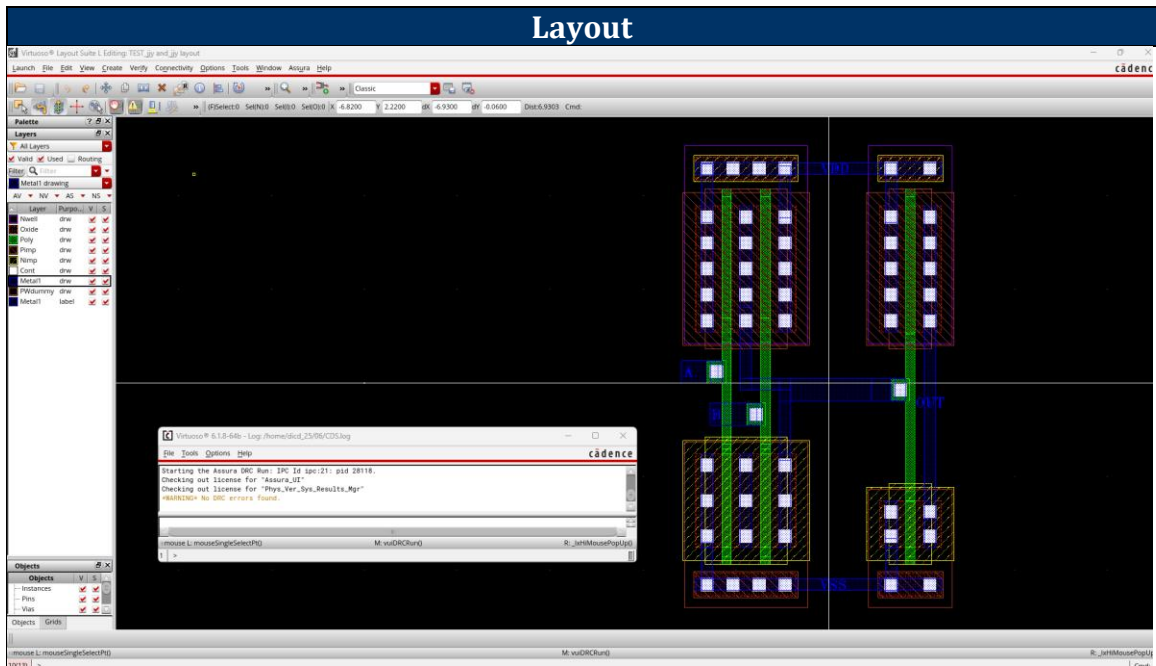
### Schematic



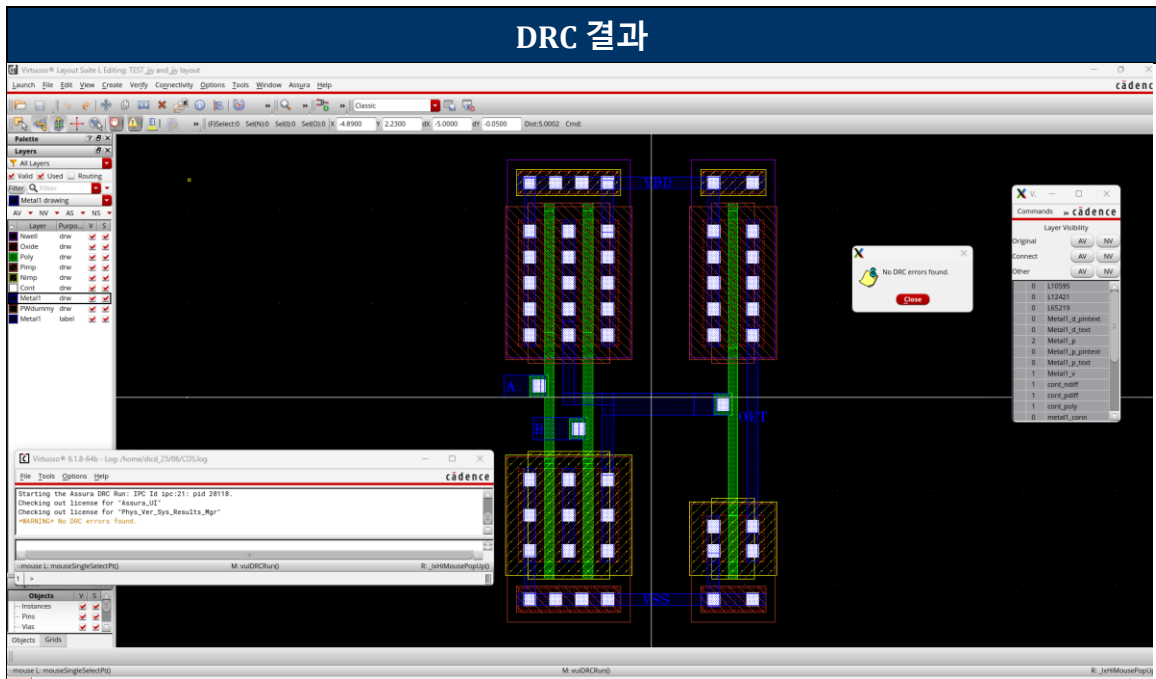
### Pre-Sim 결과(NN)



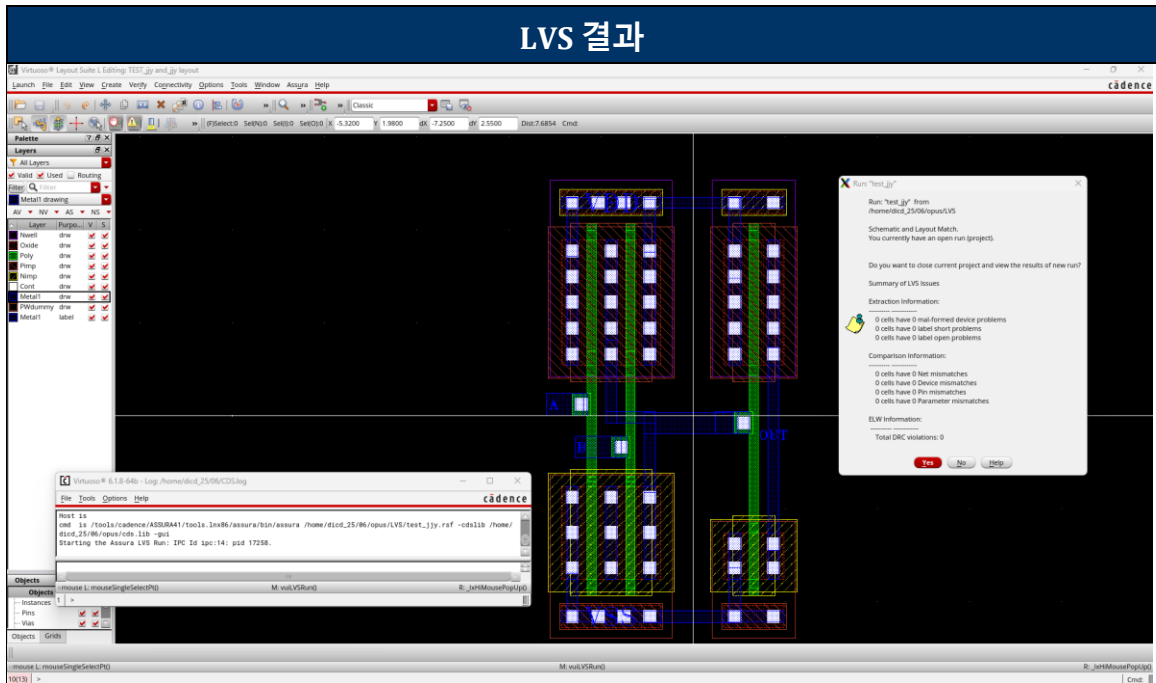
## Layout



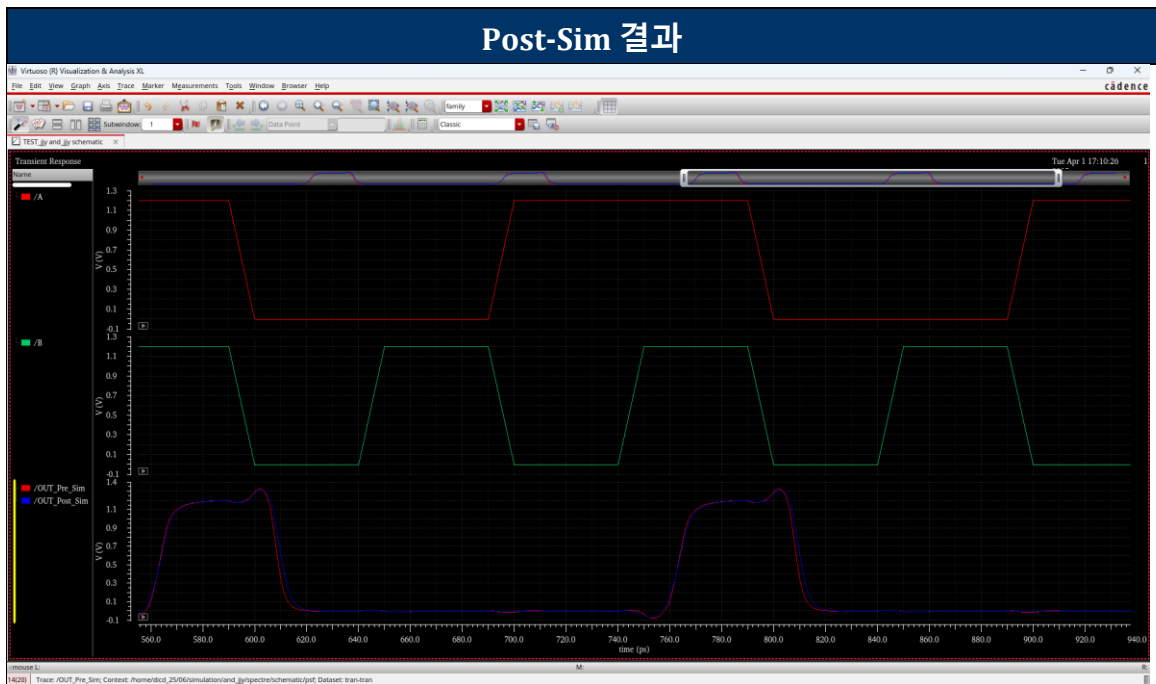
## DRC 결과



## LVS 결과

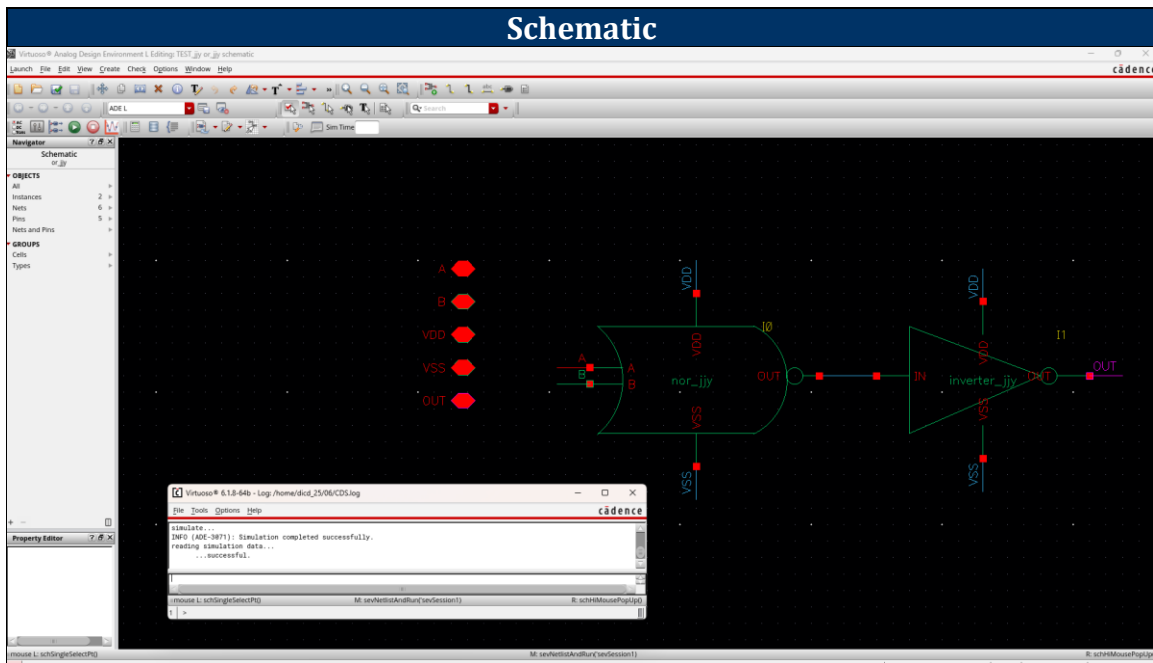


## Post-Sim 결과



## 2-4) 실습 4: OR

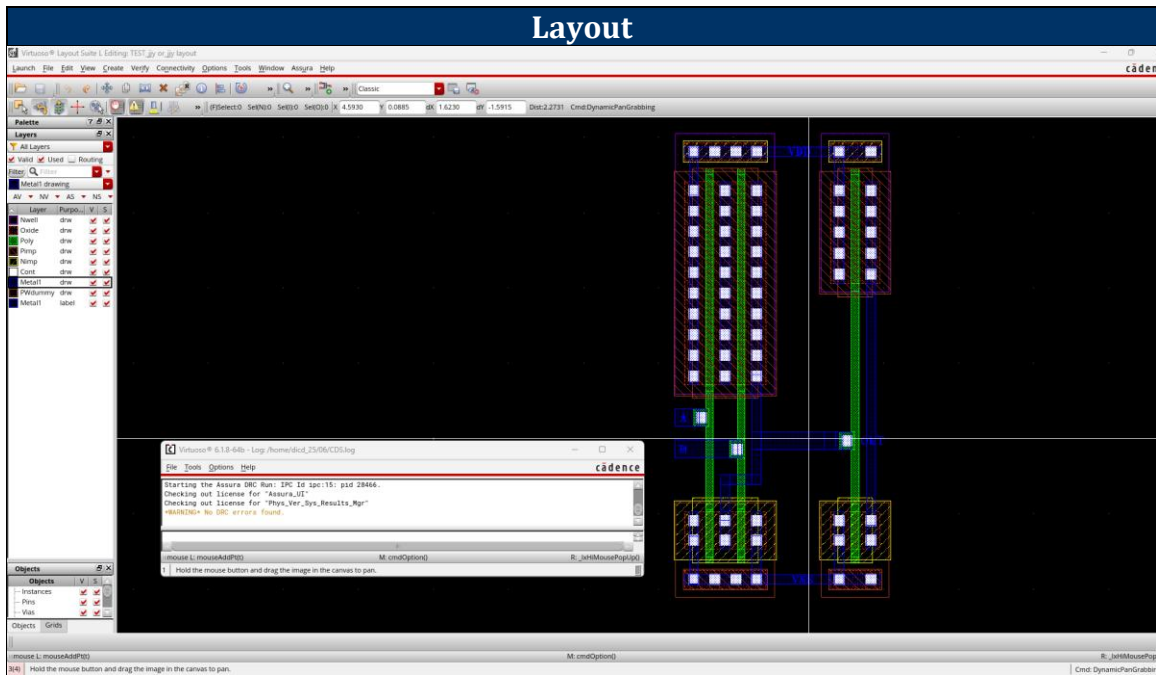
### Schematic



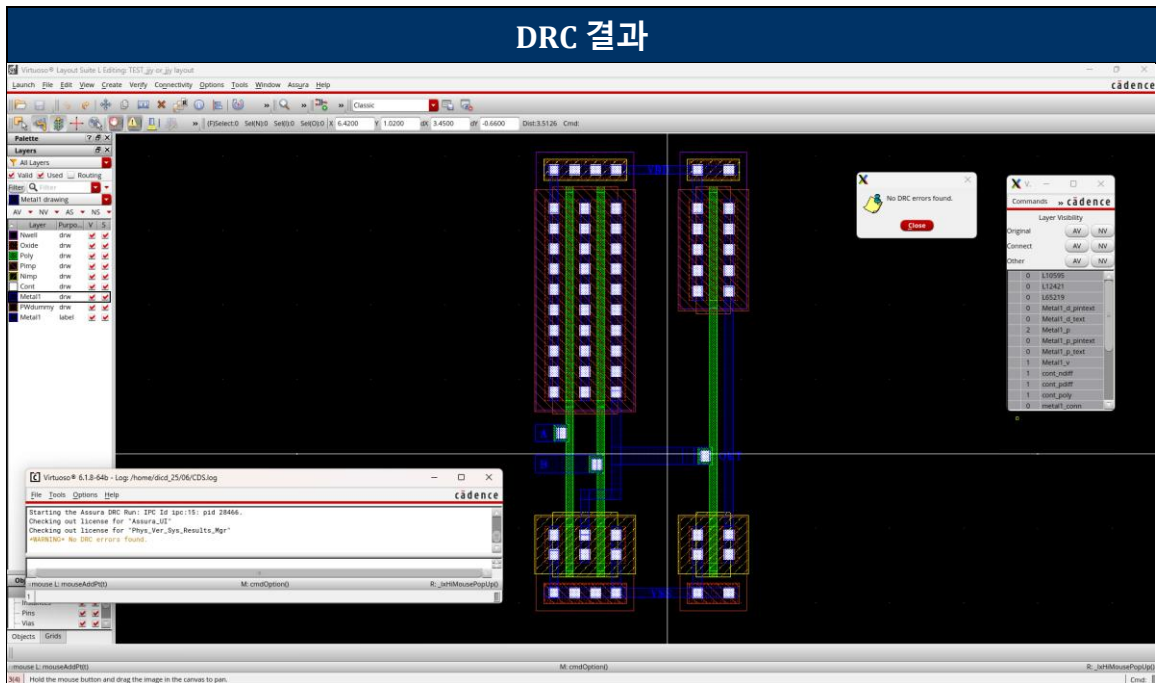
### Pre-Sim 결과(FF, NN, SS)



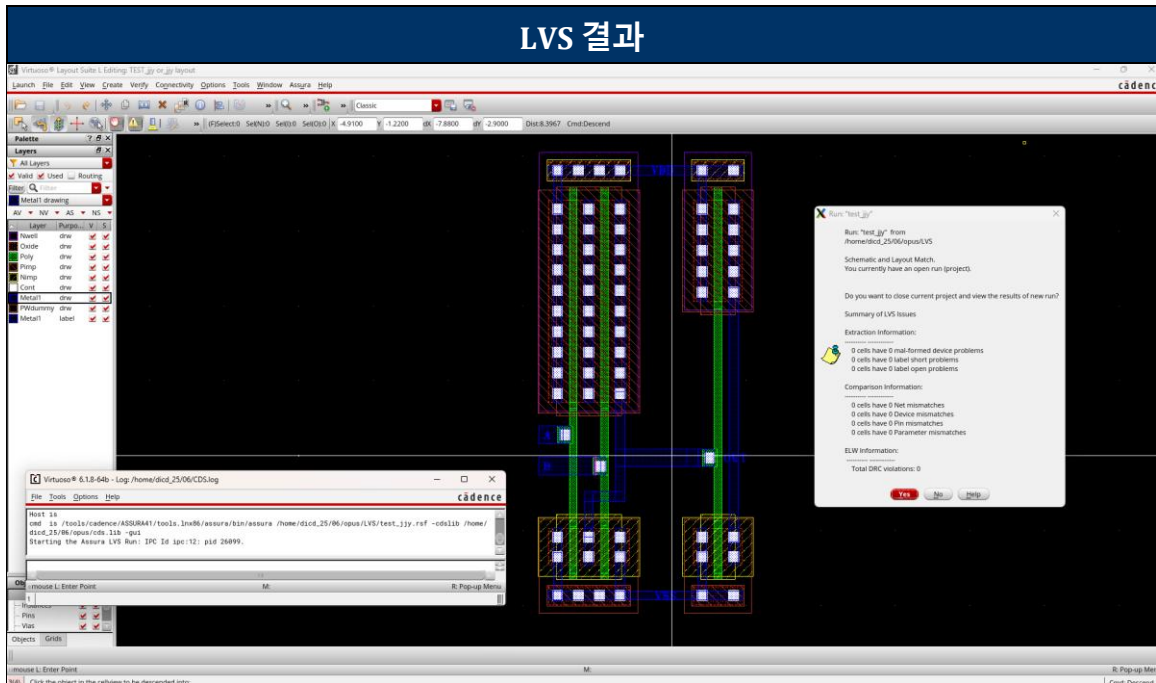
## Layout



## DRC 결과



## LVS 결과



## Post-Sim 결과

### Post-Sim 결과

진행 안함.

### 3. 실습 결론 및 고찰

이번 실습에서는 CMOS 기반의 NAND, NOR, AND, OR 논리 회로를 설계하고, 공정 조건(Process Corner) 및 시뮬레이션 환경에 따른 동작 특성의 차이를 분석하였다.

#### #1. 공정 조건에 따른 논리회로 시뮬레이션 결과 차이 (FF/NN/SS)

시뮬레이션에서 공정 조건으로 FF(Fast-Fast), NN(Nominal-Nominal), SS(Slow-Slow)를 설정하여 각각의 조건에서 논리 회로가 어떻게 동작하는지 비교하였다.

- **FF 조건**에서는 트랜지스터의 이동도가 높고 구동 능력이 커서 전체 회로의 응답 속도가 빨랐다. NAND, NOR 등의 출력 지연이 가장 적게 발생했고, 상승/하강 시간도 짧았다.
- **SS 조건**에서는 트랜지스터의 구동 능력이 약해 회로 동작이 상대적으로 느려졌고, 출력 전압이 안정적으로 도달하기까지 시간이 더 걸렸다.
- **NN 조건**은 이 둘의 중간 수준으로, 평균적인 동작 특성을 나타냈다.

특히 AND, OR 처럼 여러 트랜지스터가 직렬 또는 병렬로 연결되는 구조에서는 각 공정 조건에 따라 지연 차이가 더욱 명확하게 나타났다. 이를 통해, 동일한 회로라도 제조 공정 편차에 따라 실제 동작이 달라질 수 있으며, robust 한 회로 설계를 위해 다양한 corner case 를 고려해야 한다는 점을 체감할 수 있었다.

#### #2. Pre-Simulation vs Post-Simulation 차이

또한, 동일한 논리 회로에 대해 Pre-simulation 과 Post-simulation 을 비교하여 parasitic 요소의 영향을 분석하였다.

- **Pre-sim** 은 schematic 회로 기준의 이상적인 동작만을 반영한 시뮬레이션으로, 각 게이트의 출력 전압이 빠르게 전환되었고, 지연이 거의 없는 깨끗한 파형을 보였다.
- 반면, **Post-sim** 에서는 layout 에서 추출한 parasitic RC 정보가 포함되기 때문에, 배선 저항(R)과 기생 정전용량(C)의 영향으로 인해 전파 지연이 증가하였고, 출력의 rise/fall time 이 느려지는 현상이 관찰되었다.