
PLL 강의

Cadence Virtuoso 사용법

조교 조성근

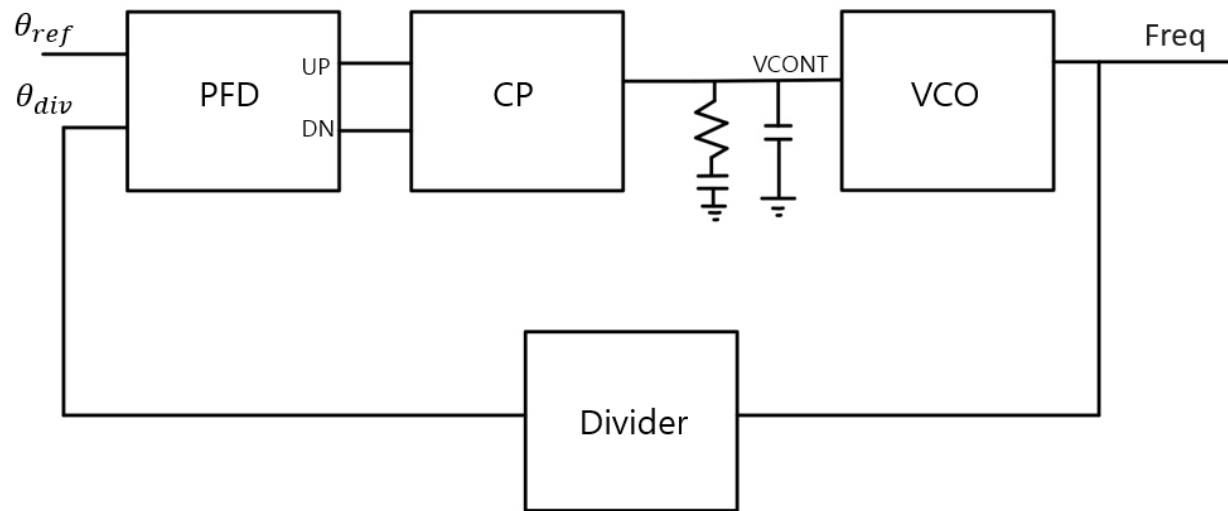
개요

- PLL
 - PLL 이란
 - PLL 원리
 - SPEC
- Cadence Virtuoso 사용법
 - 단축키
 - Library / Cell 만들기
 - Inverter 만들기
 - Simulation
 - Instance 만들기
 - 과제

PLL이란

- PLL (Phase Locked Loop) : 위상 동기 루프
 - Reference clock을 입력으로 받아 원하는 주파수의 clock을 출력
 - Reference clock은 crystal oscillator를 사용하여 생성. Noise가 거의 없는 깨끗한 clock.
 - But, crystal oscillator로는 높은 주파수의 clock 생성이 어려움 (보통 100MHz 이하)

- Top Block Diagram



- ① PFD (Phase Frequency Detector)
 - 출력과 입력 clock의 phase 및 frequency 비교

- ② CP (Charge Pump)
 - 비교 출력에 따른 전류 크기 출력

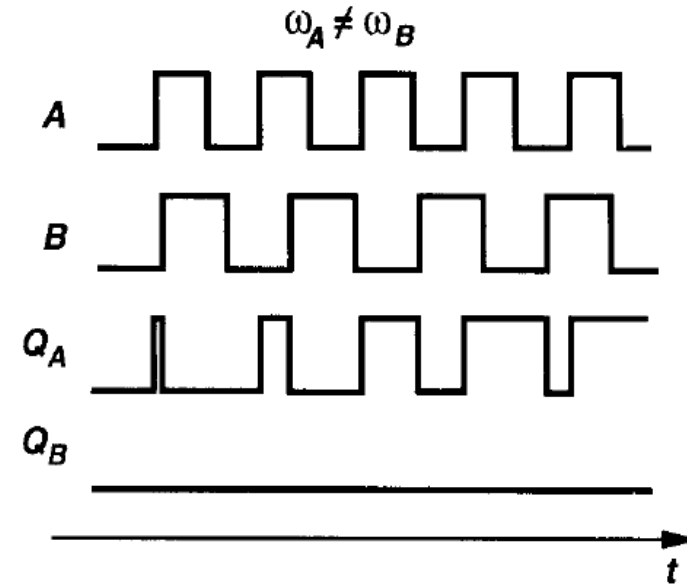
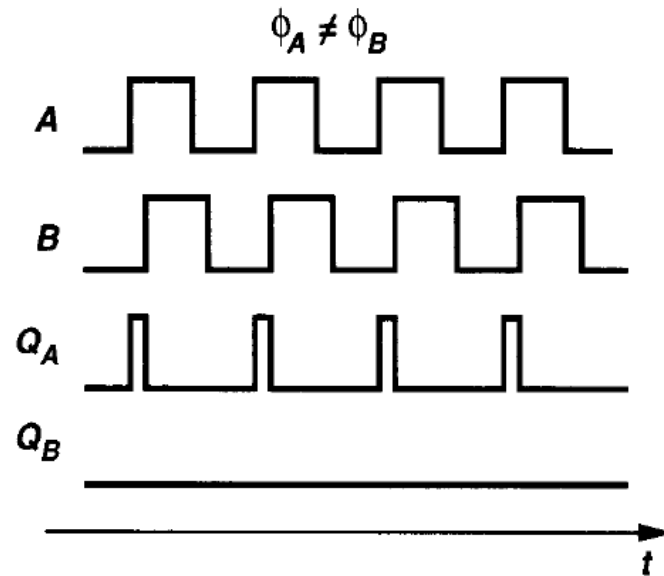
- ③ Loop Filter
 - 안정성을 고려하여 전류를 전압으로 전환

- ④ VCO (Voltage Controlled Oscillator)
 - 전압에 따른 clock 생성

- ⑤ Divider
 - 출력 clock을 저주파로 전환

PLL 원리

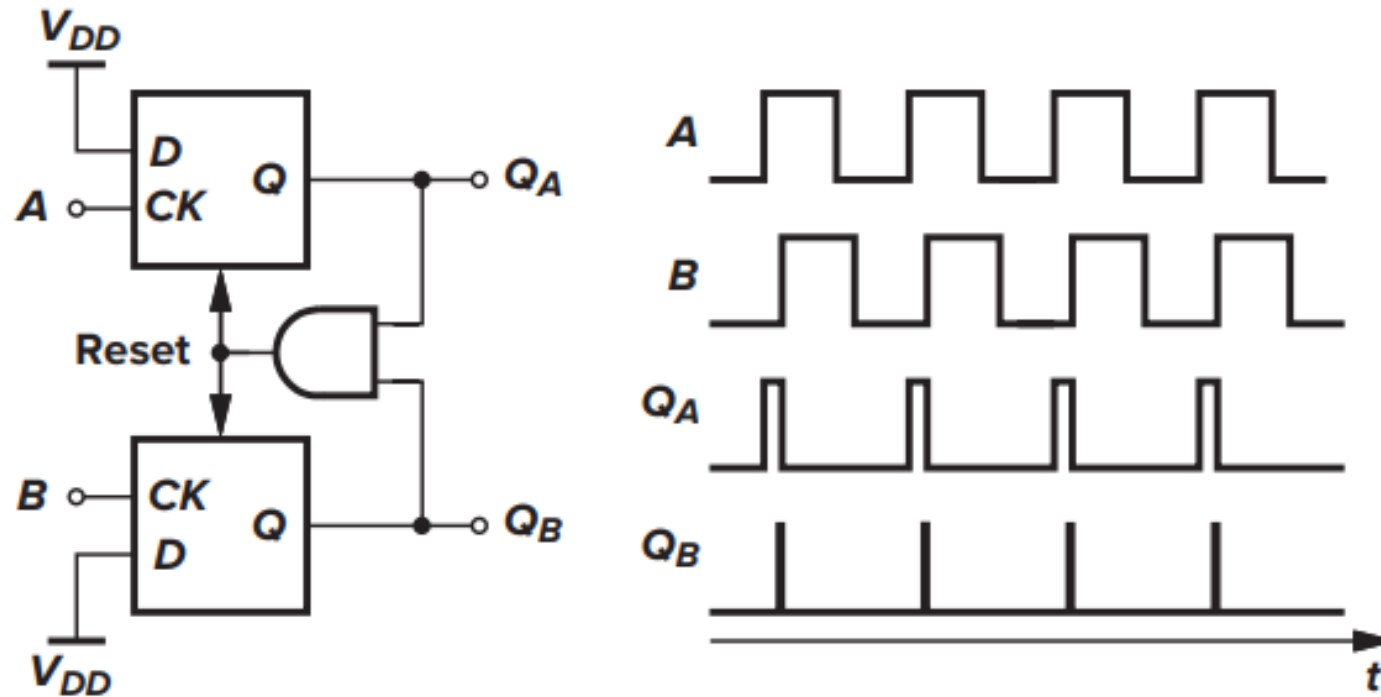
- 위상 동기(phase locking)
 - VCO의 출력 위상을 기준 위상에 정렬시키는 것
 - 위상차가 시간에 따라 변하지 않는다면 루프는 동기화 되었다고 정의한다
 - $\frac{d\phi_{out}}{dt} - \frac{d\phi_{in}}{dt} = 0 \rightarrow \omega_{out} = \omega_{in}$



PLL 구성

① PFD (Phase Frequency Detector)

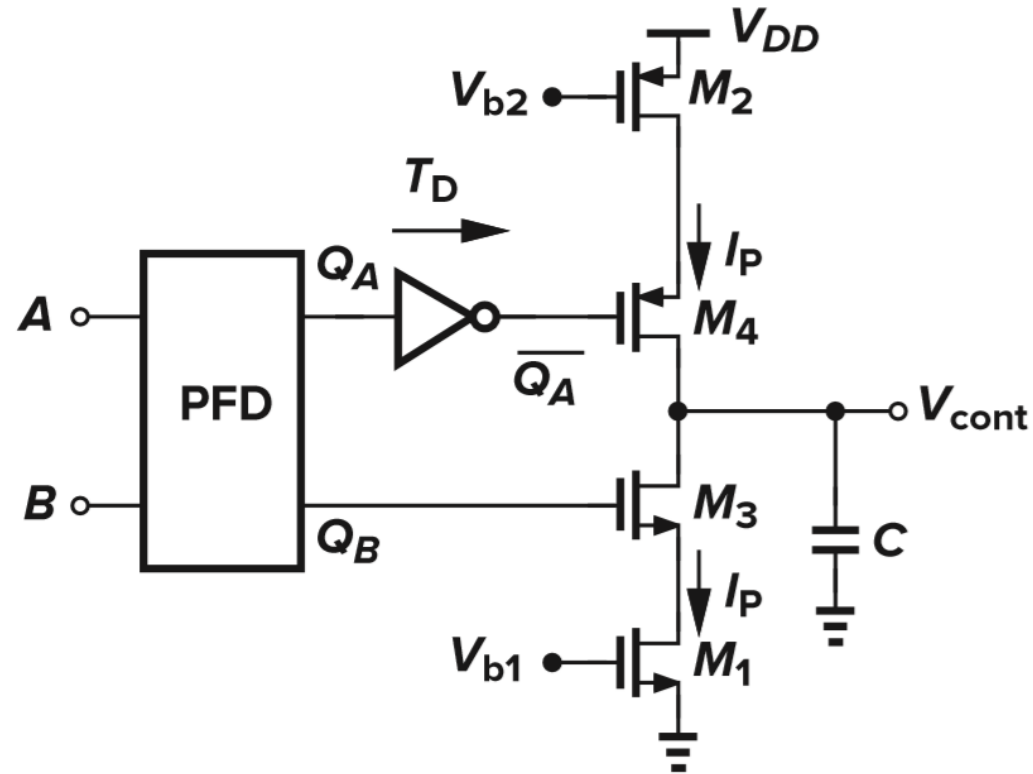
- 출력과 입력 clock의 phase 및 frequency 비교



PLL 구성

② CP (Charge Pump)

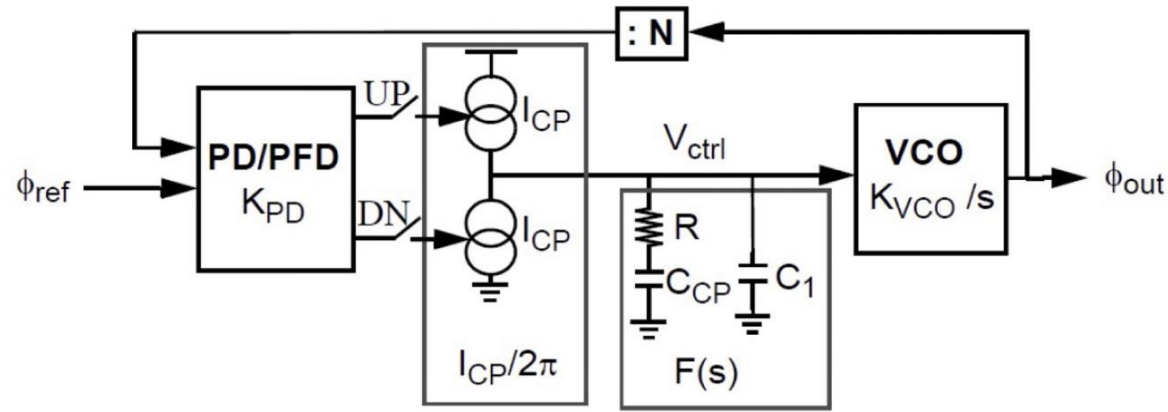
- 비교 출력에 따른 전류 크기 출력



PLL 구성

③ Loop Filter

- 안정성을 고려하여 전류를 전압으로 전환



$$H_{open}(s) = K_{PFD} \cdot I_{CP}/2\pi \cdot F(s) \cdot K_{VCO}/s$$

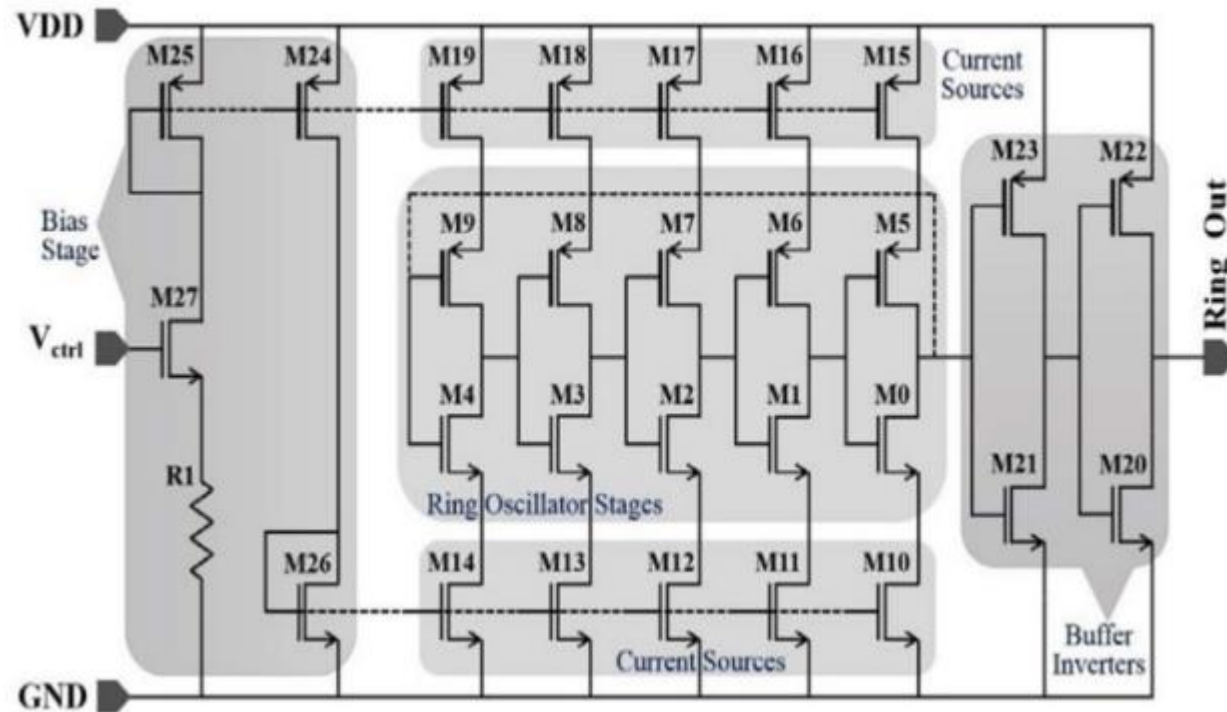
$$\text{ignoring } C_1: H_{open}(s) = K_{PFD} \cdot \frac{I_{CP}}{2\pi \cdot C_{CP}} \cdot (1 + RC_{CP}s) \cdot \frac{K_{VCO}}{s^2}$$

$$\text{with } C_1: H_{open}(s) = K_{PFD} \cdot \frac{I_{CP}}{2\pi \cdot (C_{CP} + C_1)} \cdot (1 + RC_{CP}s) \cdot \frac{K_{VCO}}{s^2 \cdot [1 + R(C_{CP} || C_1)s]}$$

PLL 구성

④ VCO (Voltage Controlled Oscillator)

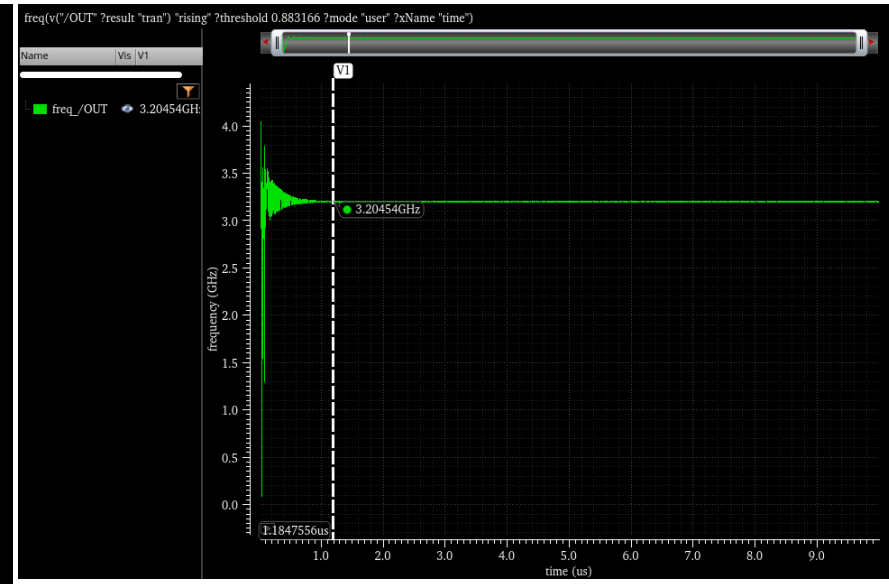
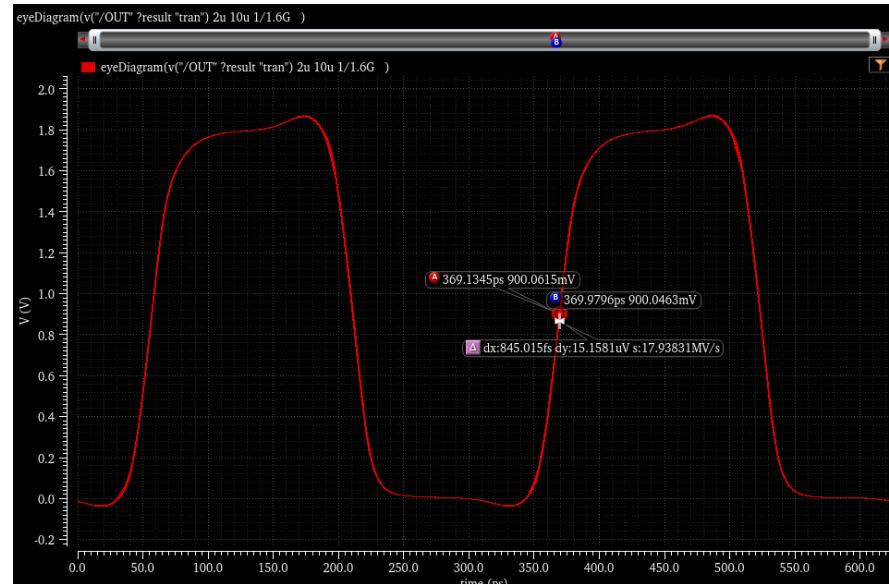
- 전압에 따른 clock 생성



SPEC

- Jitter
 - Time domain 관점에서 noise 또는 회로 안정성에 의해 주기가 흔들리는 것을 표현한 지표
 - 본 강의에서는 peak-to-peak jitter를 확인할 예정

- Power
- Lock Time (Settling Time)



Cadence Virtuoso 사용법

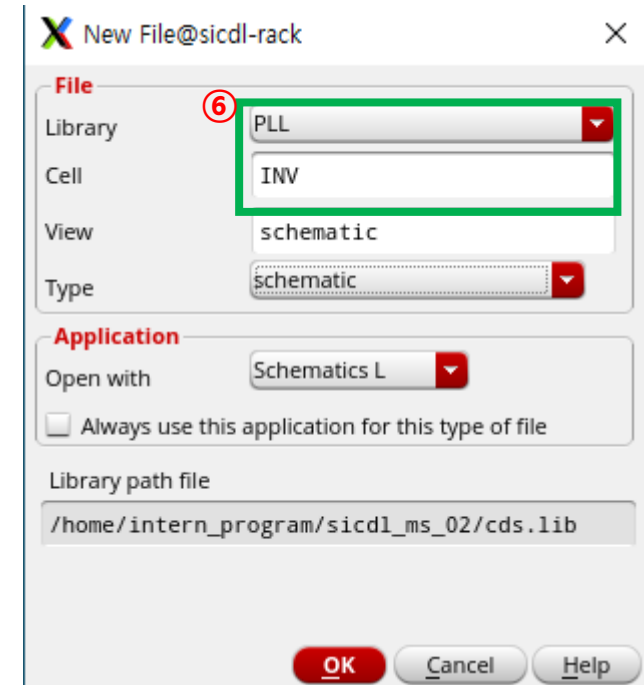
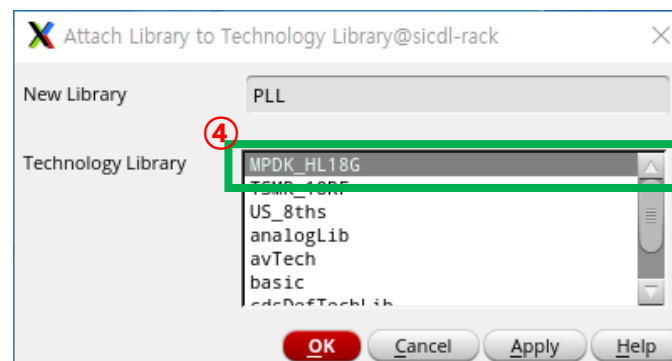
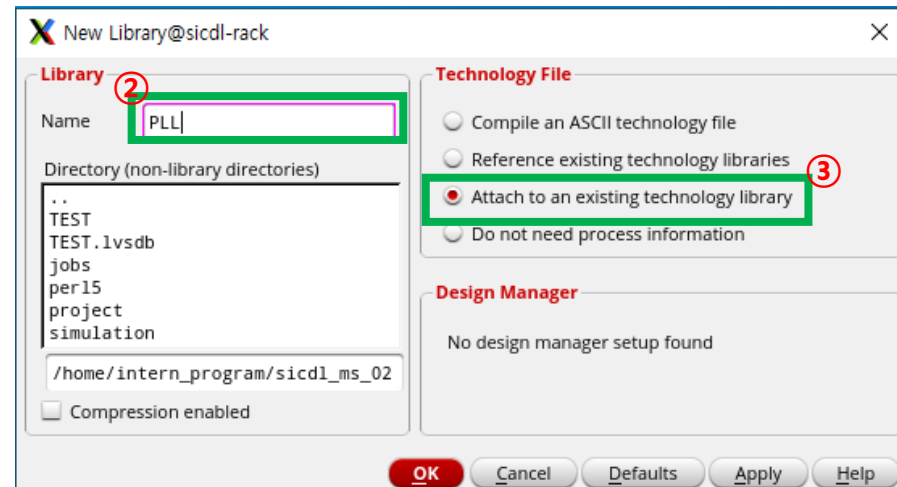
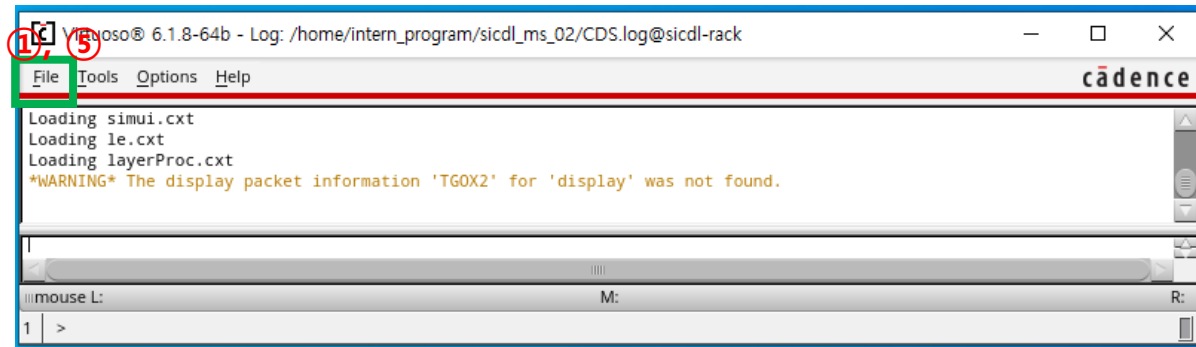
- 단축키
 - i : instance
 - w : wire
 - f : full
 - u : 뒤로 가기
 - q : 소자 특성
 - p : port 생성
 - l : label 생성
 - shift + x : 저장
 - 수업 중 필요한 단축키 추후 안내 예정

Cadence Virtuoso 사용법 (Library/Cell 만들기)

- File -> New -> Library / Cellview

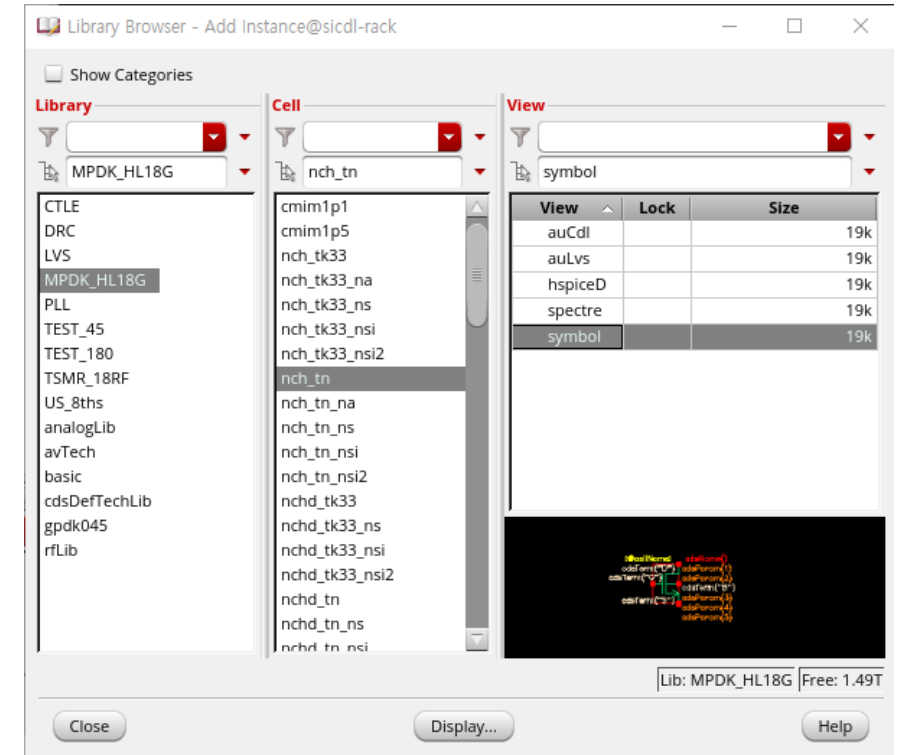
① 본인 Library 만들기 (ex. PLL)

② 원하는 Cell 만들기 (ex. INV)

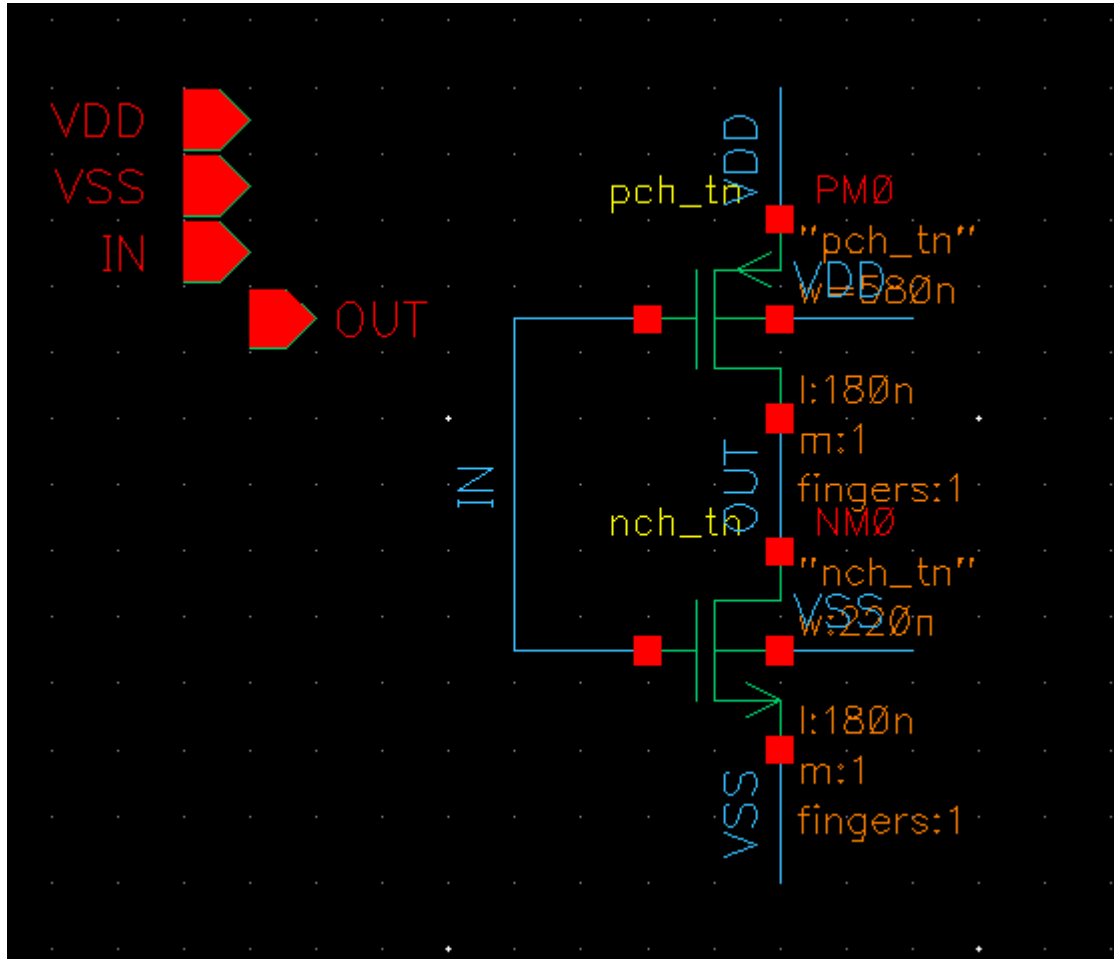


Cadence Virtuoso 사용법 (Library/Cell 만들기)

- i : instance 생성
- Mosfet / 저항/ cap 등을 사용 가능
- MPDK_HL18G
 - NMOS : nch_tn
 - PMOS : pch_tn
- analogLib
 - CAP : cap
 - RES : res
 - 그 외 : 수업 중 필요한 소자 추후 안내



Cadence Virtuoso 사용법 (Inverter 만들기)



- Inverter 만들기

- ① 소자 배치 (단축키 i)
- ② wire 연결 (단축키 w)
- ③ labeling (단축키 l)
- ④ port 지정 (단축키 p)
- ⑤ 저장 (단축키 shift+x)

- NMOS : PMOS width 비

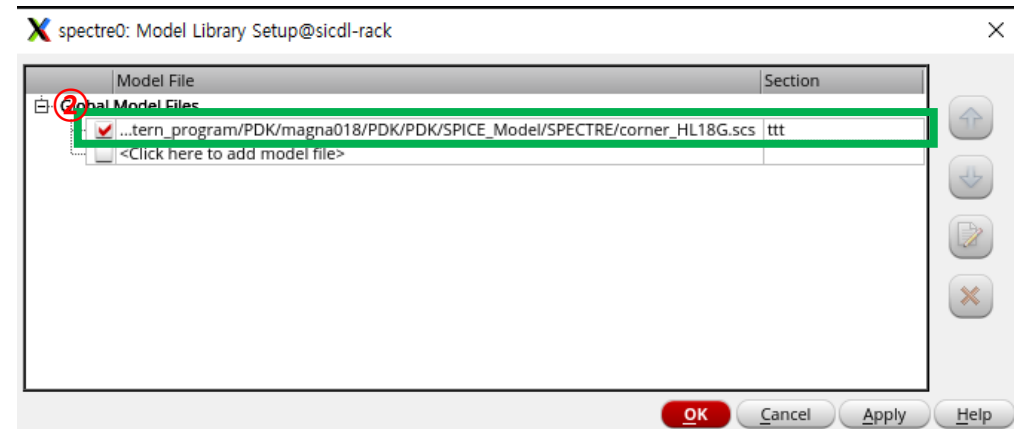
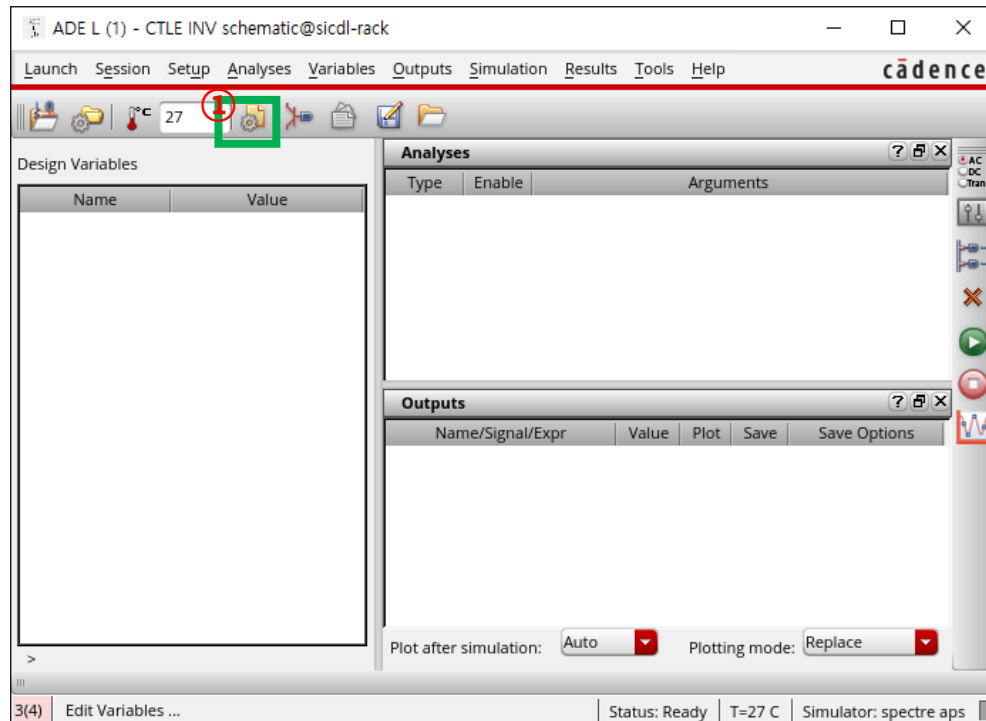
= 220n : 580n

-> MOSFET mobility 비율 고려

-> 공정마다 해당 비율은 상이함

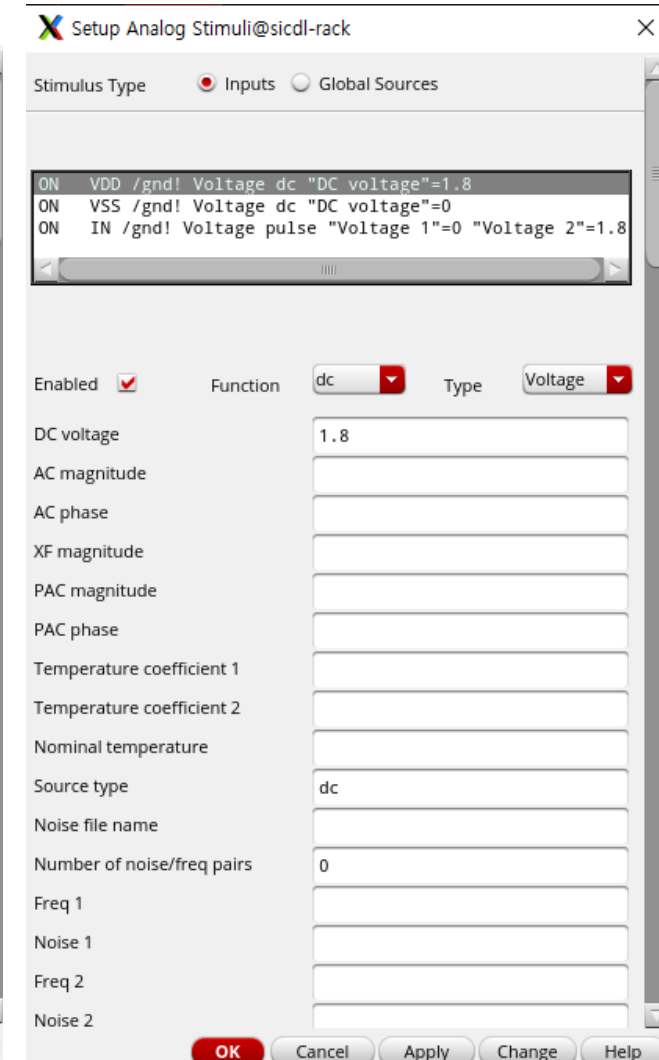
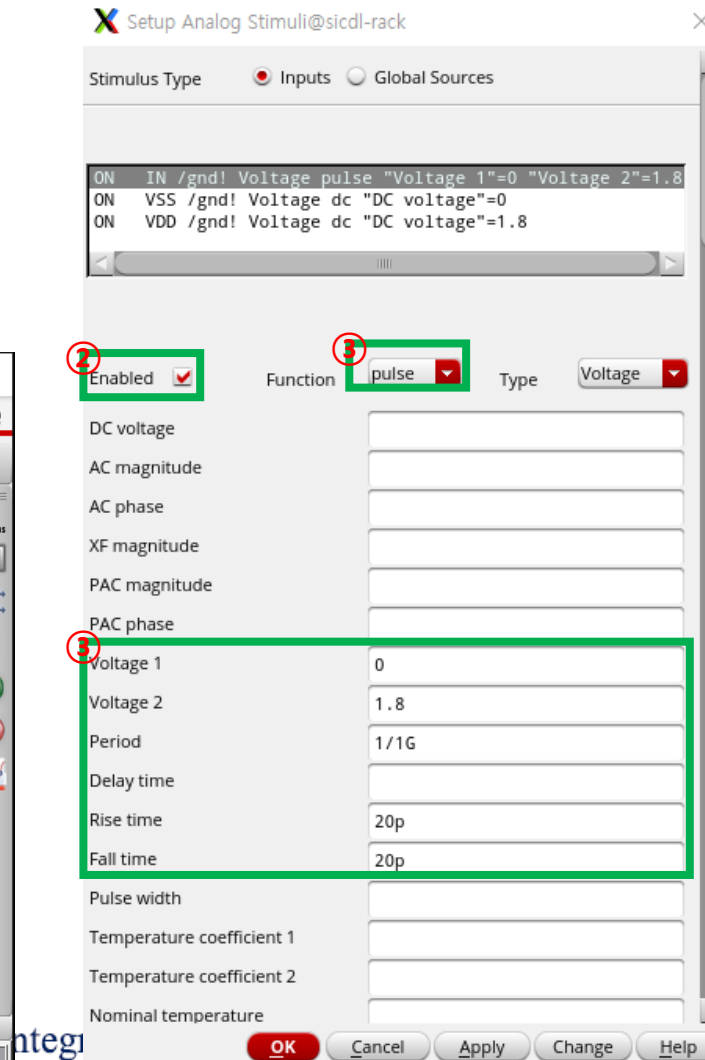
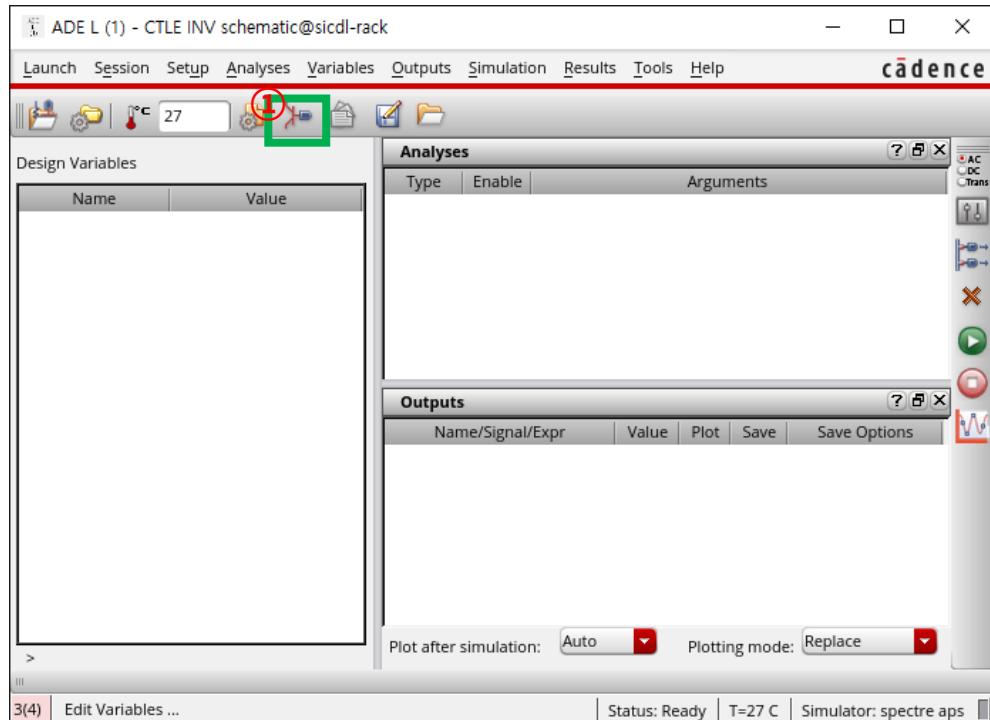
Cadence Virtuoso 사용법 (Simulation)

- Launch -> ADE L
 - 공정 물리기
 - > 경로 : /home/intern_program/PDK/magna018/PDK/PDK/SPICE_Model/SPECTRE/corner_HL18G.scs
 - Corner Simulation : ttt



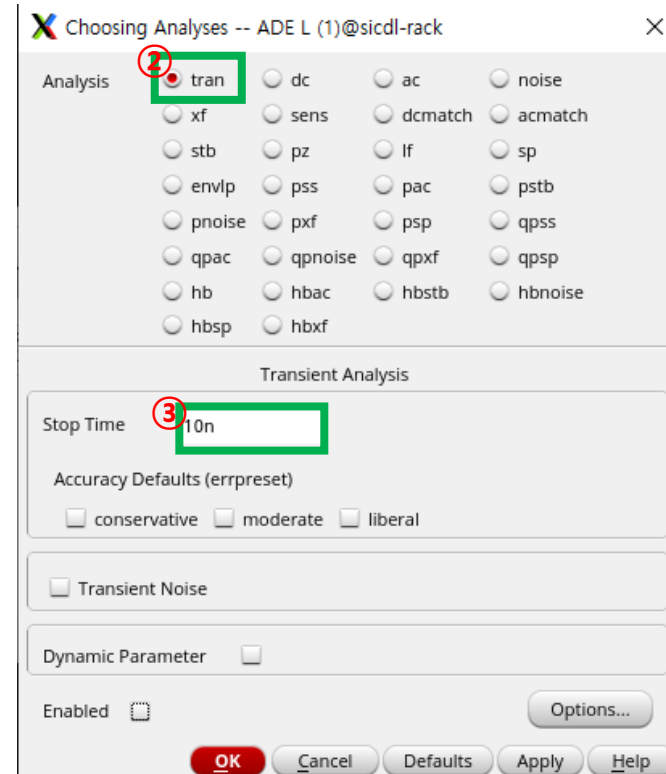
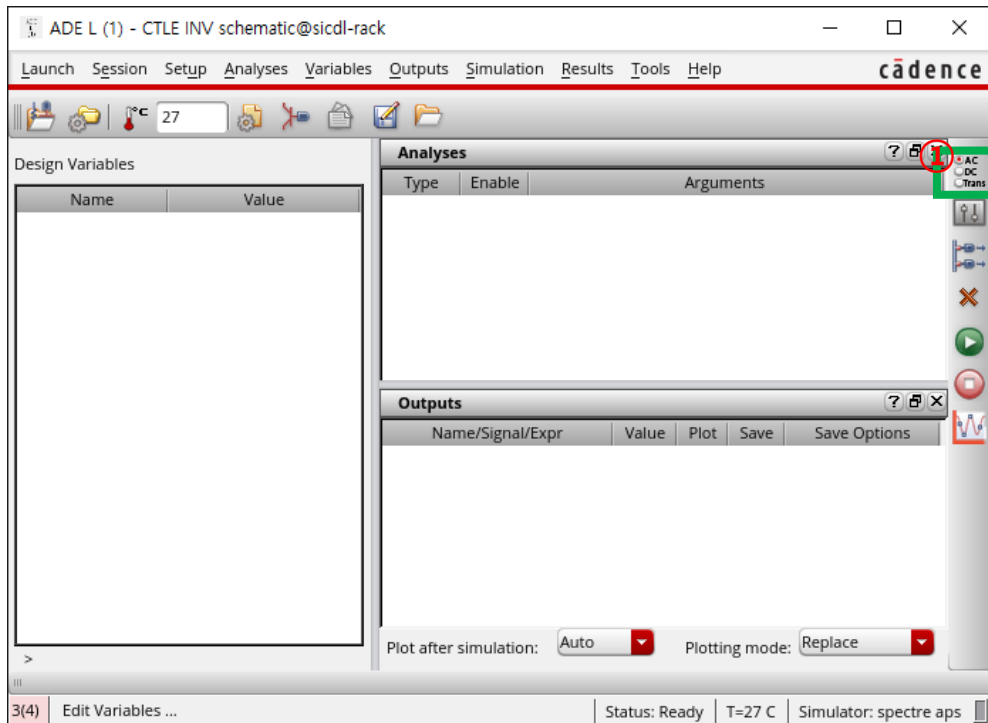
Cadence Virtuoso 사용법 (Simulation)

- 입력 신호 설정
 - 해당 공정 공급 전압 : 1.8V
 - VDD, VSS : DC 신호
 - IN : pulse 파 입력
 -



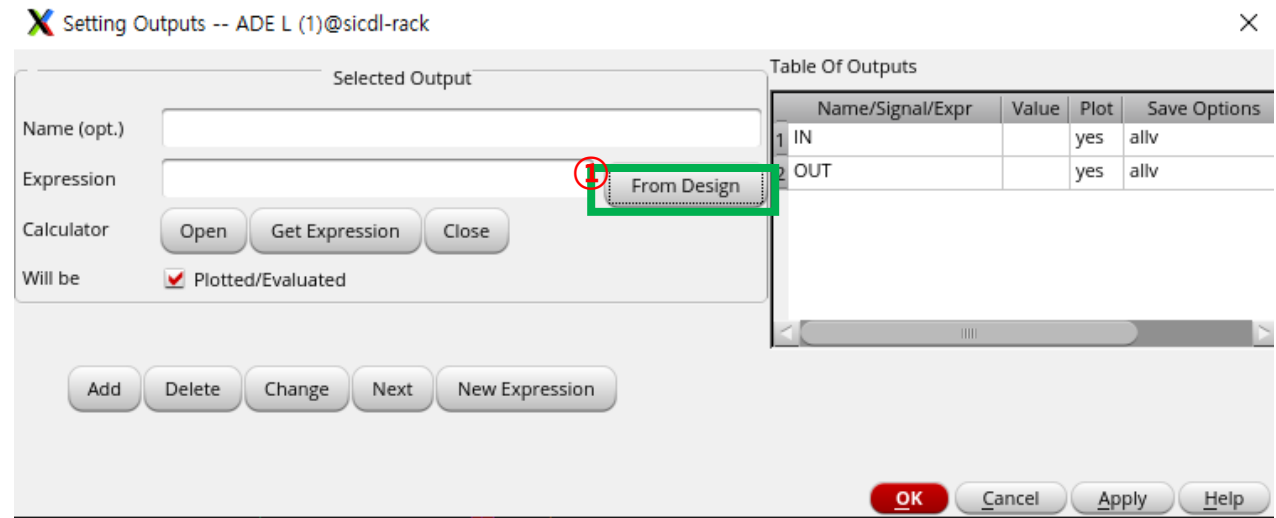
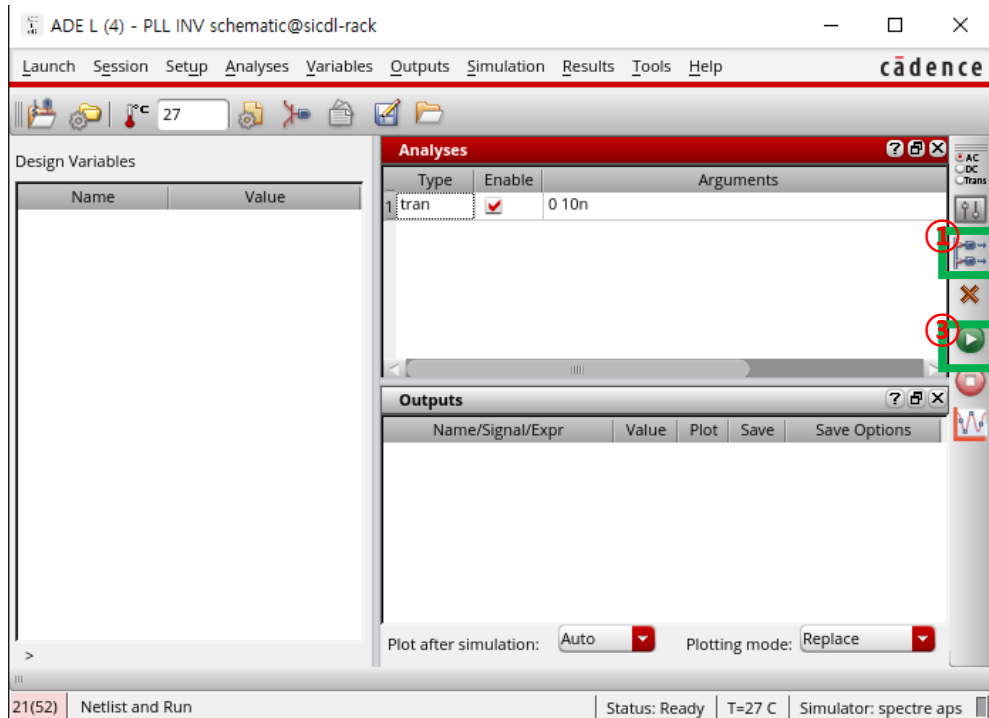
Cadence Virtuoso 사용법 (Simulation)

- 필요한 analyses 선택
 - transition simulation : 10ns 동안 시뮬레이션
 - 수업 중 필요한 analyses 방법 추후 안내 예정



Cadence Virtuoso 사용법 (Simulation)

- Simulation 출력 선택
 - From Design 버튼 누른 후 schematic 에서 원하는 signal 선택
 - 선택 완료 후 꼭 esc 버튼 눌러주기
- > 위 절차를 모두 완료 후 재생 버튼 누르면 simulation 시작!!



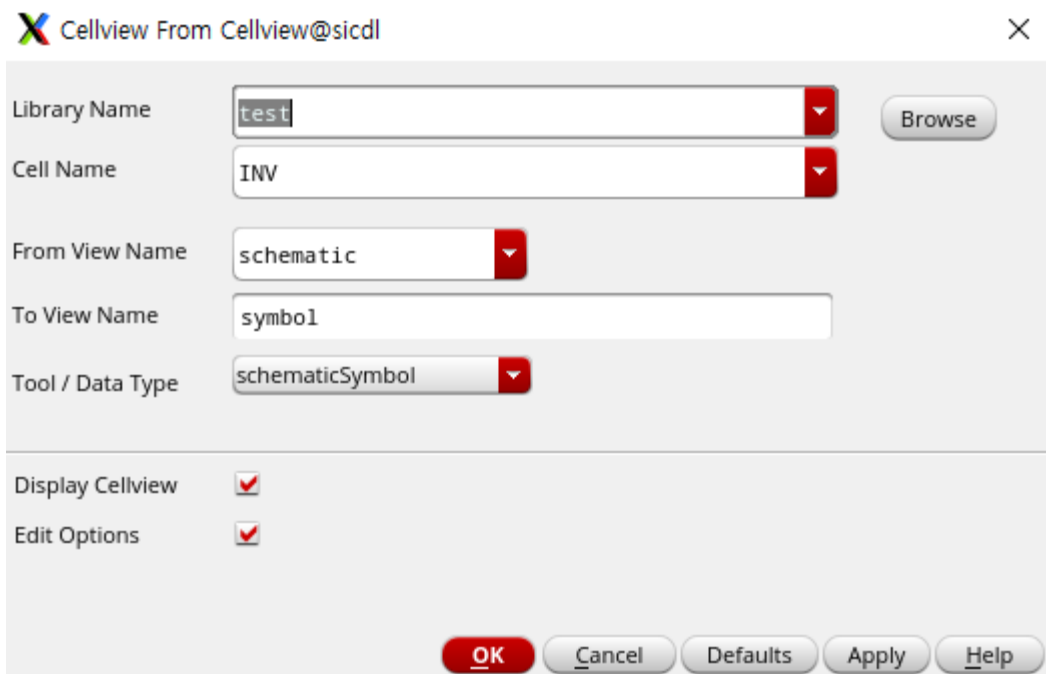
Cadence Virtuoso 사용법 (Inverter 만들기)

- Simulation 결과



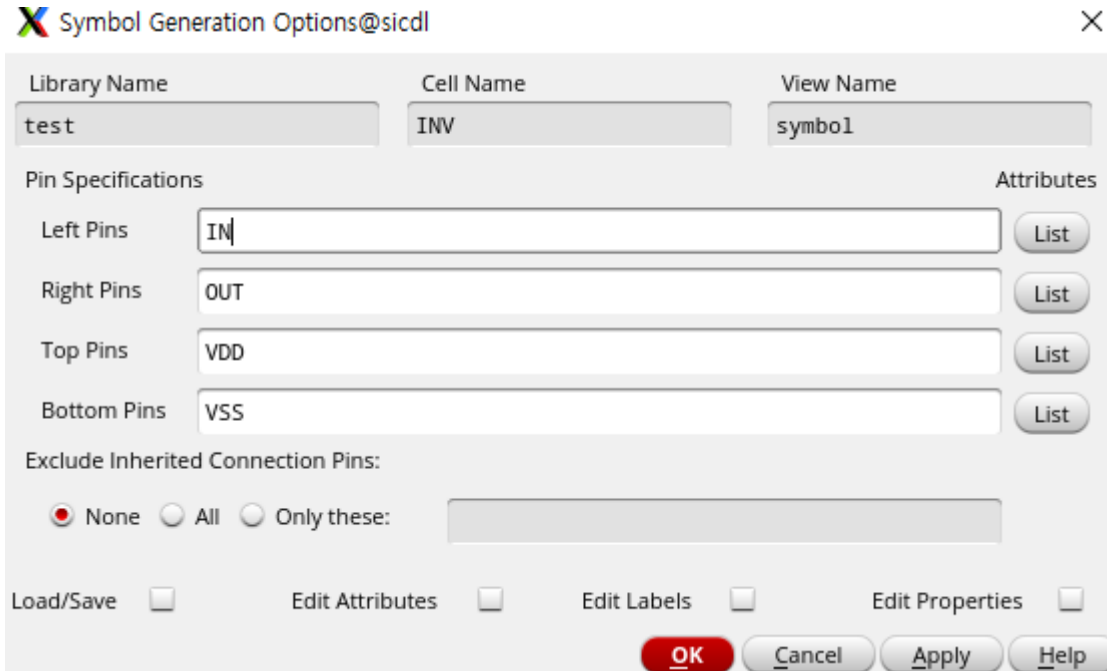
Instance 만들기

- Create -> Cellview -> From Cellview



OK 누르고 다음 창

Instance 만들기

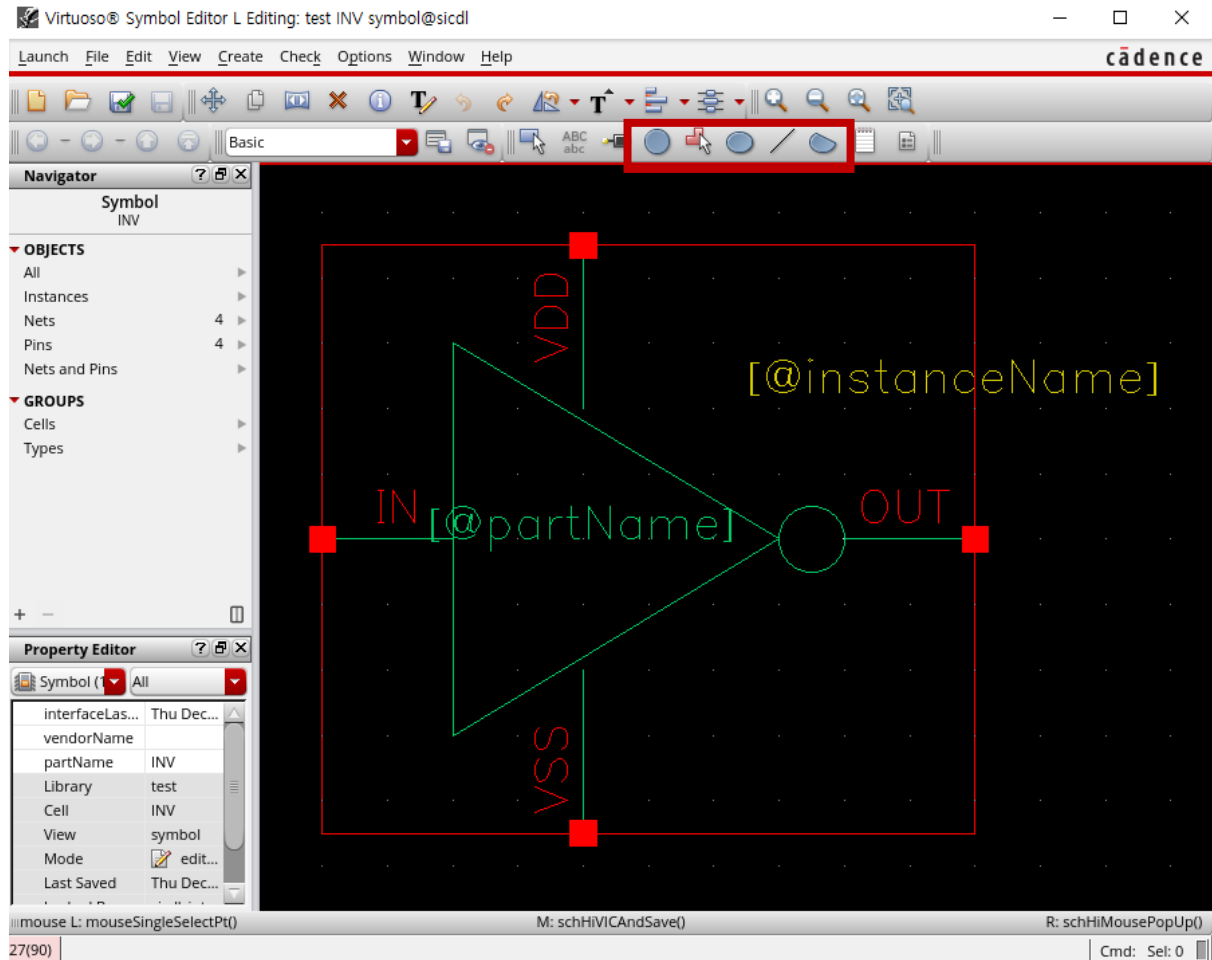


The image shows a 'Symbol Generation Options' dialog box for a component named 'INV'. The 'Library Name' is 'test' and the 'View Name' is 'symbol'. Under 'Pin Specifications', the pins are defined as: Left Pins: IN, Right Pins: OUT, Top Pins: VDD, and Bottom Pins: VSS. Each pin specification has a 'List' button next to it. Below this, there is a section for 'Exclude Inherited Connection Pins' with radio buttons for 'None' (selected), 'All', and 'Only these:'. At the bottom, there are checkboxes for 'Load/Save', 'Edit Attributes', 'Edit Labels', and 'Edit Properties', all of which are currently unchecked. The bottom right corner contains four buttons: 'OK' (highlighted in red), 'Cancel', 'Apply', and 'Help'.

Field	Value
Library Name	test
Cell Name	INV
View Name	symbol
Left Pins	IN
Right Pins	OUT
Top Pins	VDD
Bottom Pins	VSS
Exclude Inherited Connection Pins	None

- PIN의 위치를 원하는 방향에 배치
- 주로 VDD는 Top Pins에 VSS는 Bottom Pins에 배치
- 위치를 모두 정했으면 OK

Instance 만들기

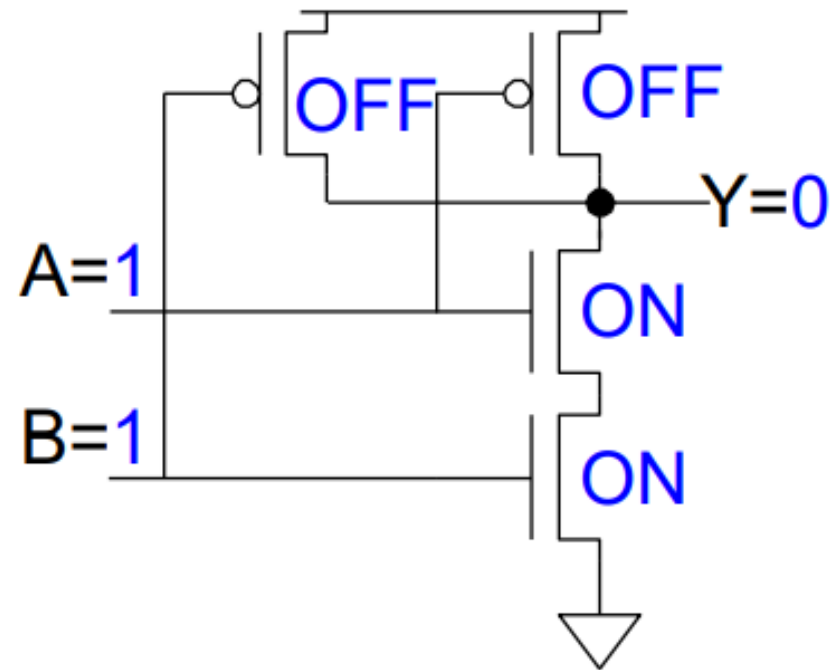
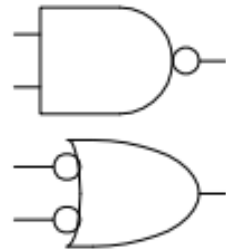


- 새롭게 드는 Tap에서 원하는 모양 및 PIN 배치 가능
- 모든 작업이 끝났으면 Shift+X로 저장
- 사이즈만 변경하였을 때는 수정 필요 없음
- PIN이 추가 및 제거되었거나 이름이 바뀌었다면 새로 Cellview 수정 필요

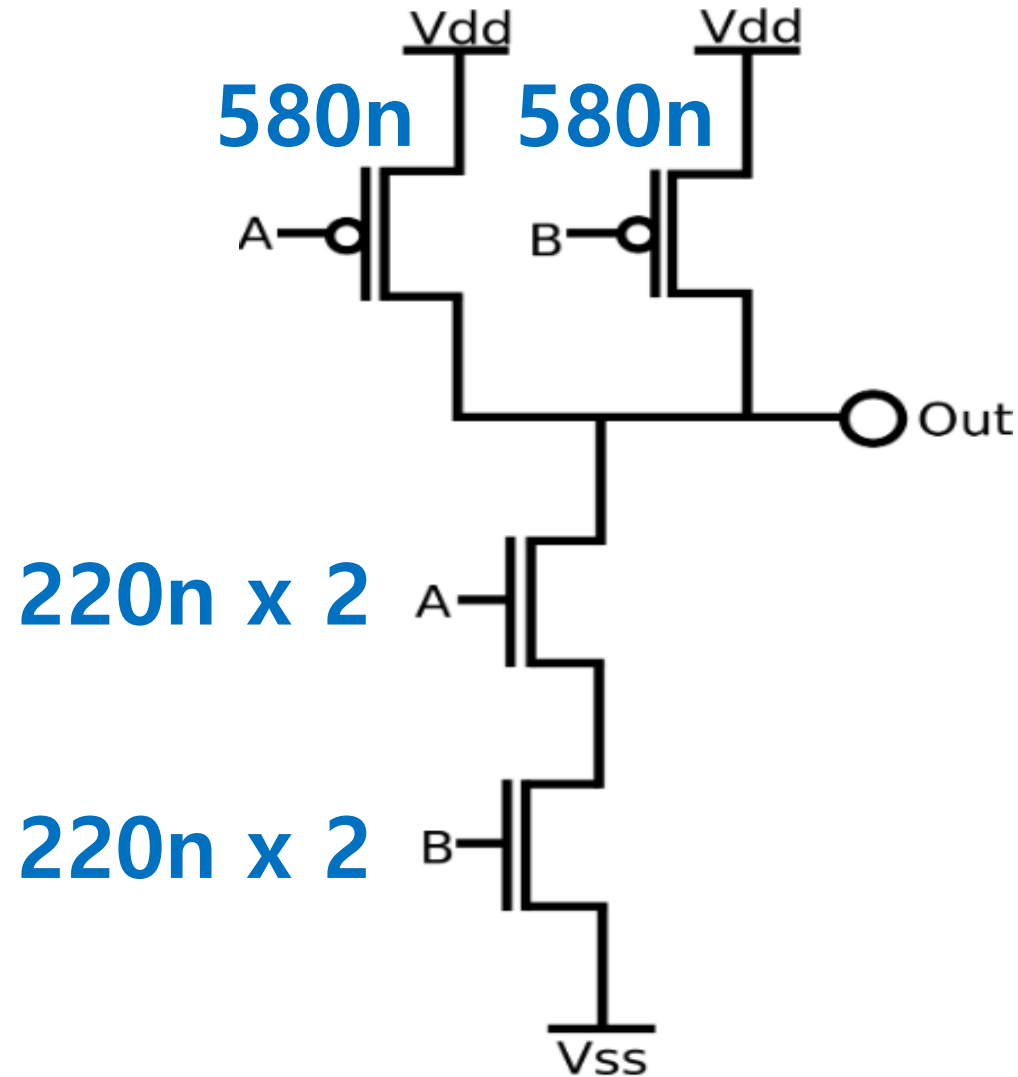
과제

- NAND Gate 만들기
- AND Gate 만들기

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



NAND Gate



- NMOS mobility가 PMOS mobility의 2배라는 전제
- 직렬 연결 Path의 size를 N배 해주어야 rising time과 falling time을 비슷한 수준으로 유지 가능