

# CAD HW4 - Modeling Mixed-Signal System and Simulating with AMS

Teaching Assistant: Hsin Tzu,Chao

Date: 12/11, 2023

Lab: ED-413

# Outline

---

- Introduction to AMS
- AMS Simulation Setup
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - Lab1: Analog Model Simulation with AMS
    - Lab2: Mixed-Signal Model Simulation with AMS
    - Lab3: 4-bit ADC with Verilog-AMS

# Outline

---

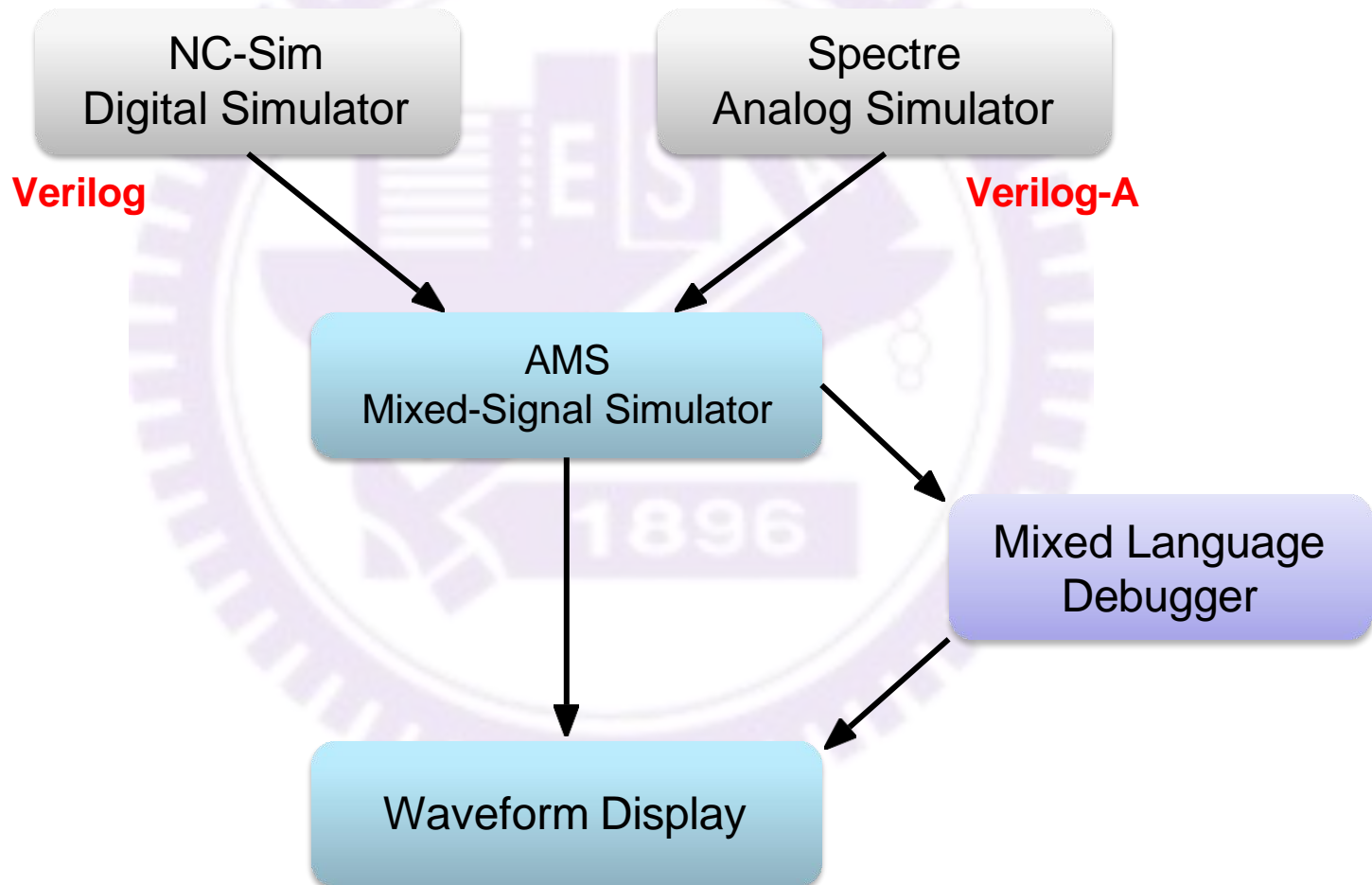
- Introduction to AMS
- AMS Simulation Setup
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - Lab1: Analog Model Simulation with AMS
    - Lab2: Mixed-Signal Model Simulation with AMS
    - Lab3: 4-bit ADC with Verilog-AMS

# What is AMS Designer

---

- Top-down system-on-chip simulation for complex mixed-signal designs
- A single executable simulator incorporating the fastest in digital and most flexible analog simulation capability
  - Digital: NC-Sim
  - Analog: Spectre
- Simulation of complex designs incorporating any and all of the following:
  - Verilog, VHDL
  - Verilog-A, Verilog-AMS, VHDL-AMS
  - Spectre
  - SPICE
  - Composer schematics

# What is AMS Designer(cont.)



# Mixed-Signal Simulation with Model

---

- The mixed-signal behavioral model simulation can verify:
  - System behavior is correct or not?
  - System requirement is met or not?
  - System performance is satisfied or not?
- Weaknesses:
  - Only time domain information can be obtained directly
    - All behavioral model should be converted into time domain
    - Other characteristics might be calculated from time domain data

# Outline

---

- Introduction to AMS
- **AMS Simulation Setup**
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - Lab1: Analog Model Simulation with AMS
    - Lab2: Mixed-Signal Model Simulation with AMS
    - Lab3: 4-bit ADC with Verilog-AMS



# Environment Setting

---

- NC-Verilog / Verilog-XL
  - e.g `> source /usr/cad/cadence/CIC/incisiv.cshrc`
- Spectre
  - e.g `> source /usr/cad/cadence/CIC/mmsim.cshrc`
- Composer / Virtuoso
  - e.g `> source /usr/cad/cadence/CIC/ic_06.17.709.cshrc`



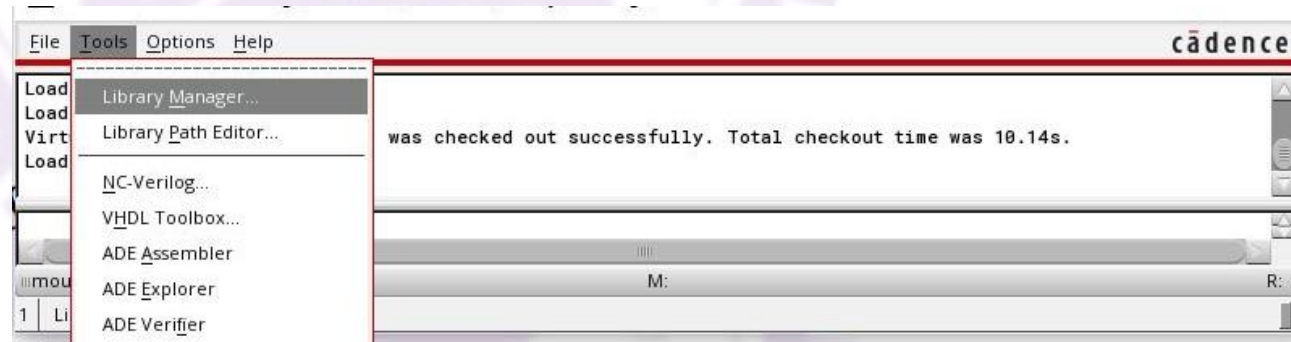
# Simulation Flow

Step 1. > virtuoso &

Step 2. Open library manager

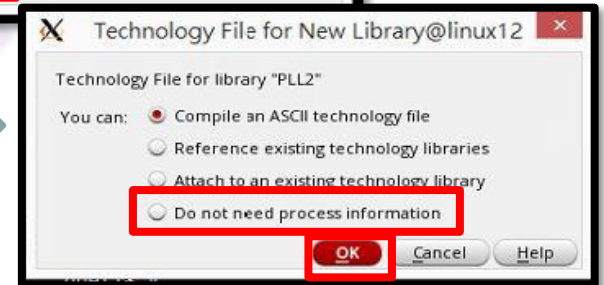
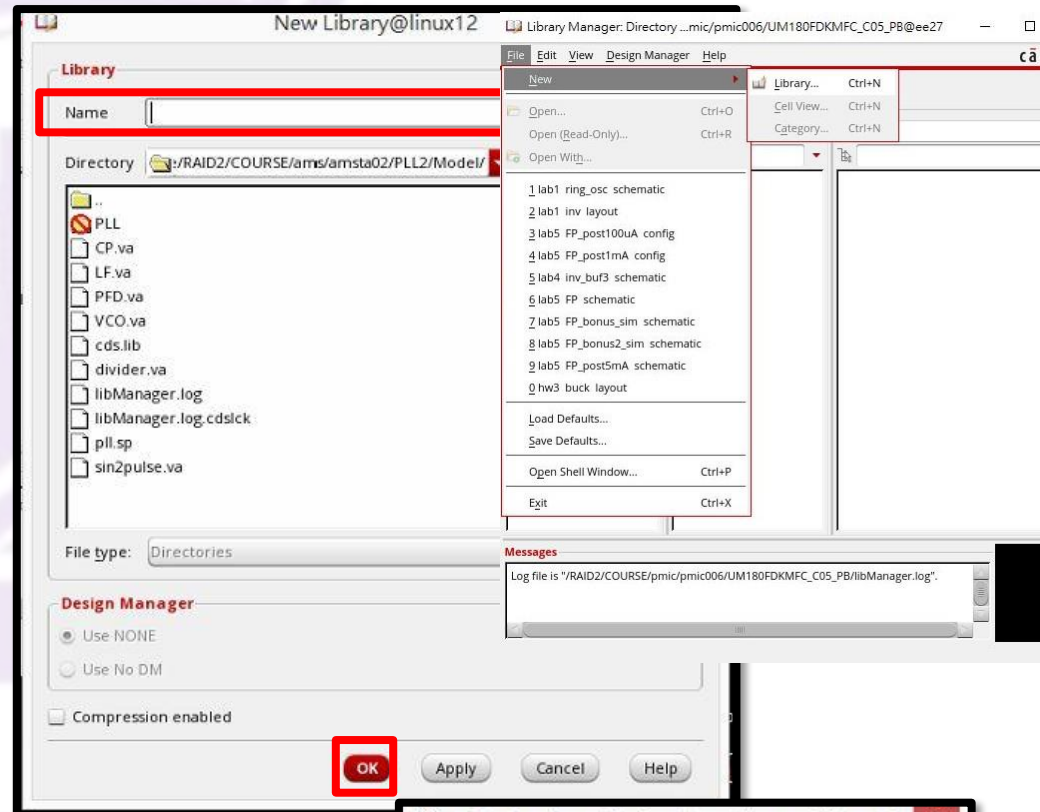
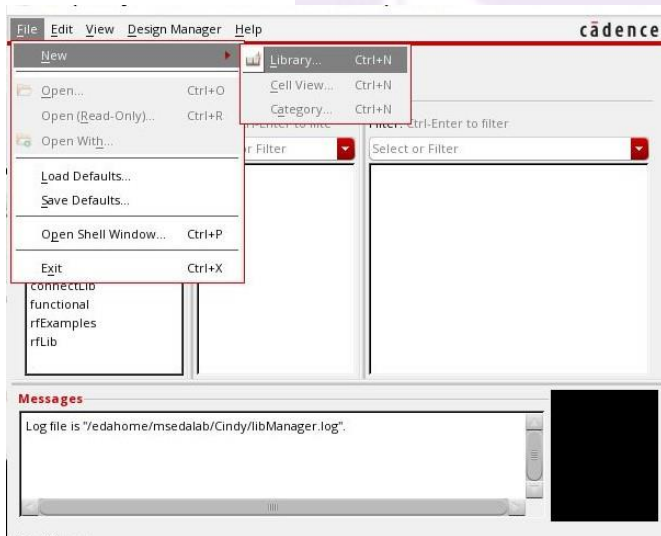
- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

Open library manager



# Simulation Flow

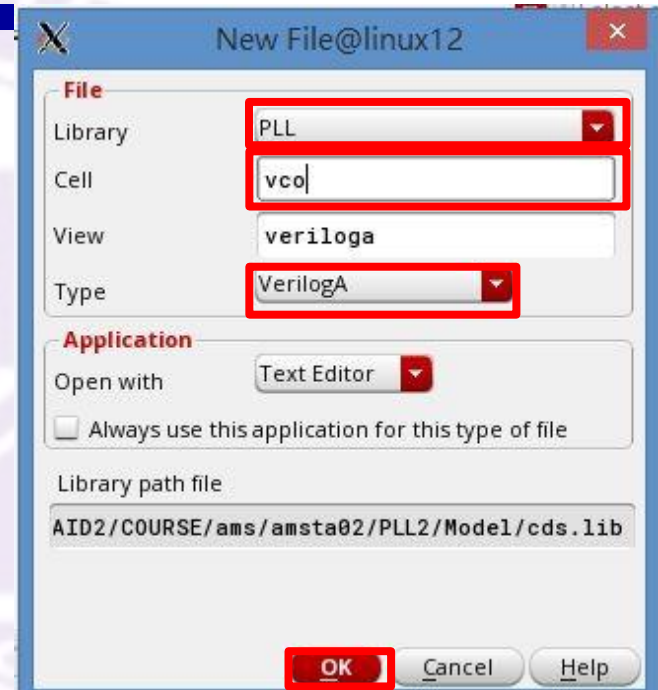
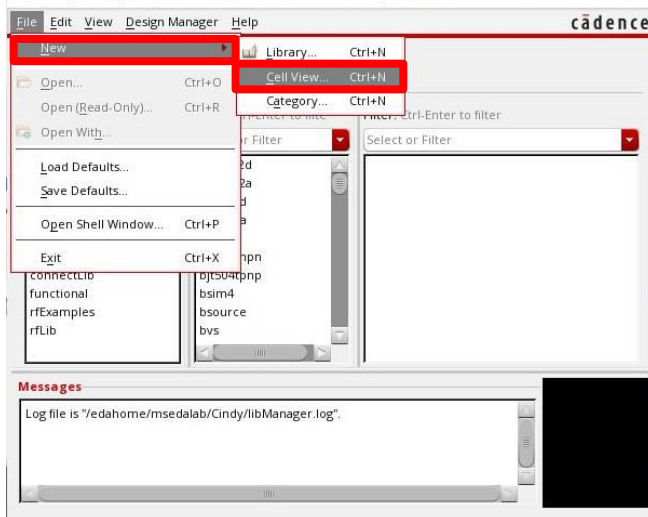
- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE



# Create Verilog-A Cells

- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

Create a new cell view

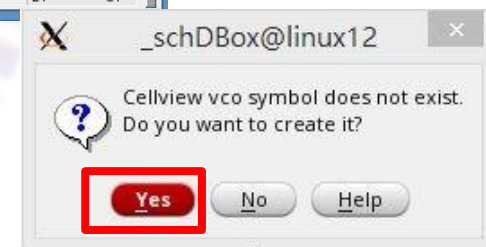
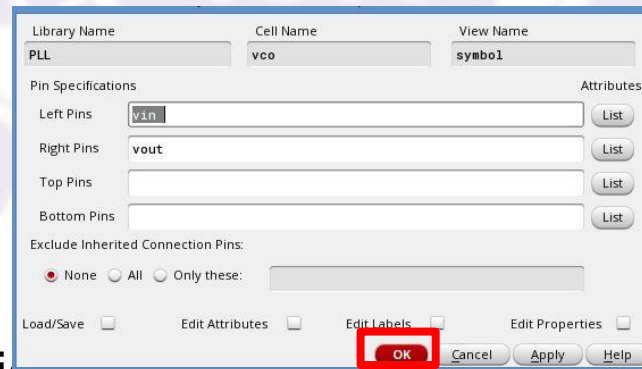
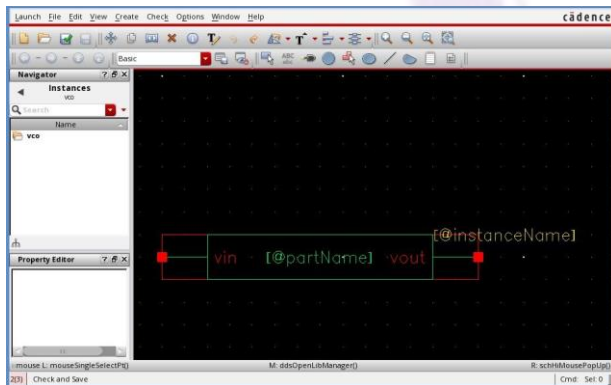
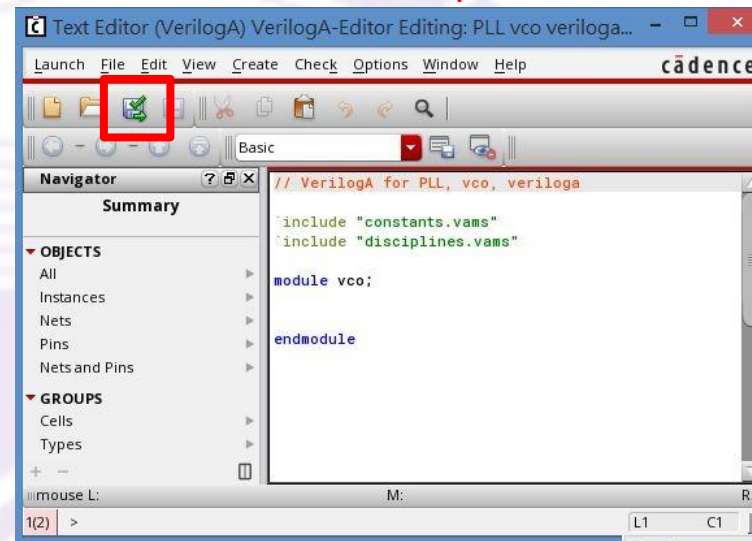


1. Choose your library
2. Input the name on the Cell Name column
3. Choose the **VerilogA** type for Analog model
4. OK

# Designing with Verilog-A

- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

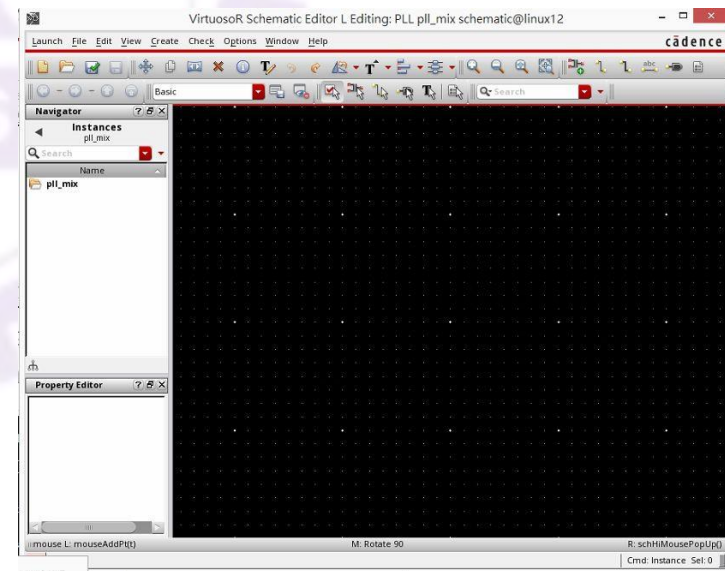
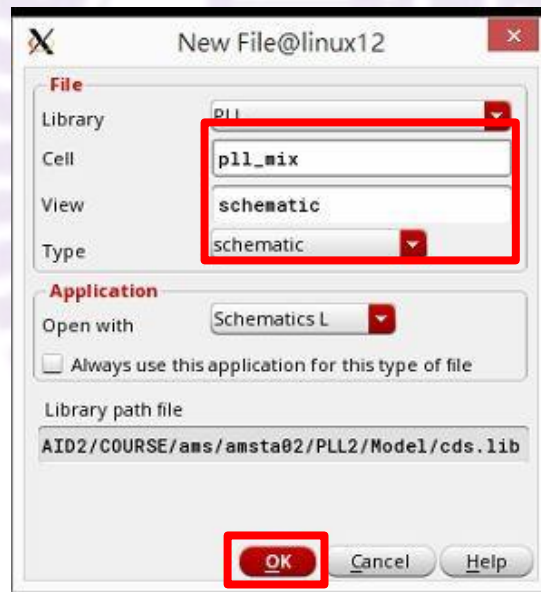
Enter behavioral description



# Create Top Cell - Schematic

- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

Input the name on the Cell Name column and choose the Schematic

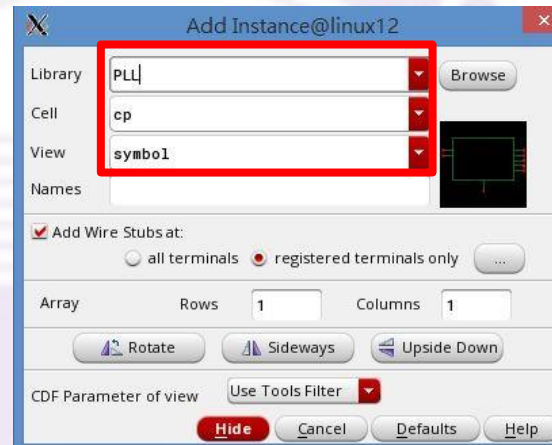




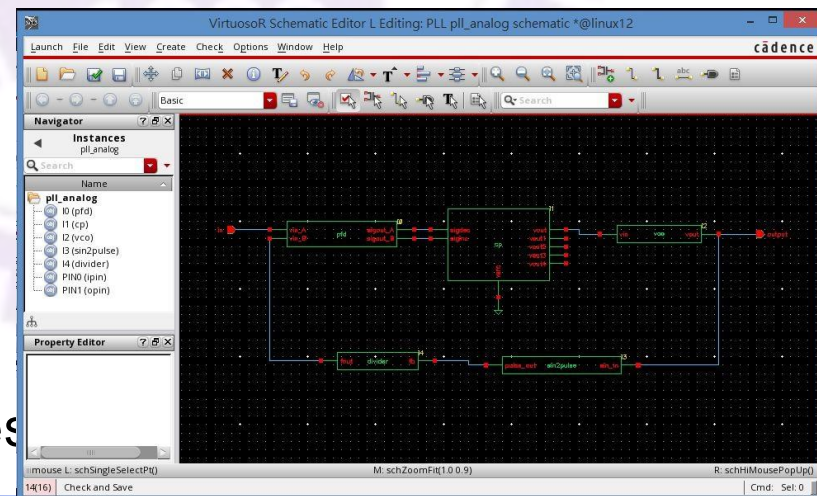
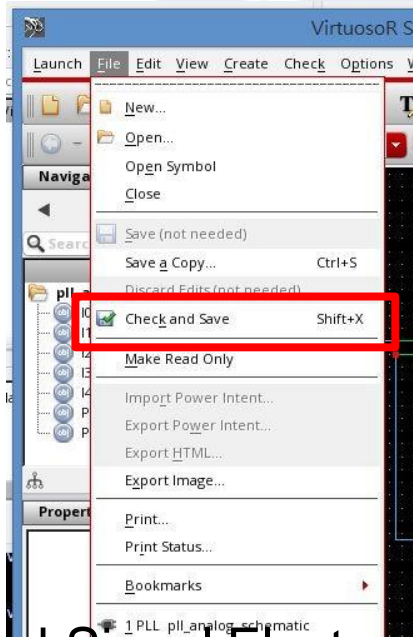
# Create Top Cell - Schematic

- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

Add Instance: Create → instance (i)

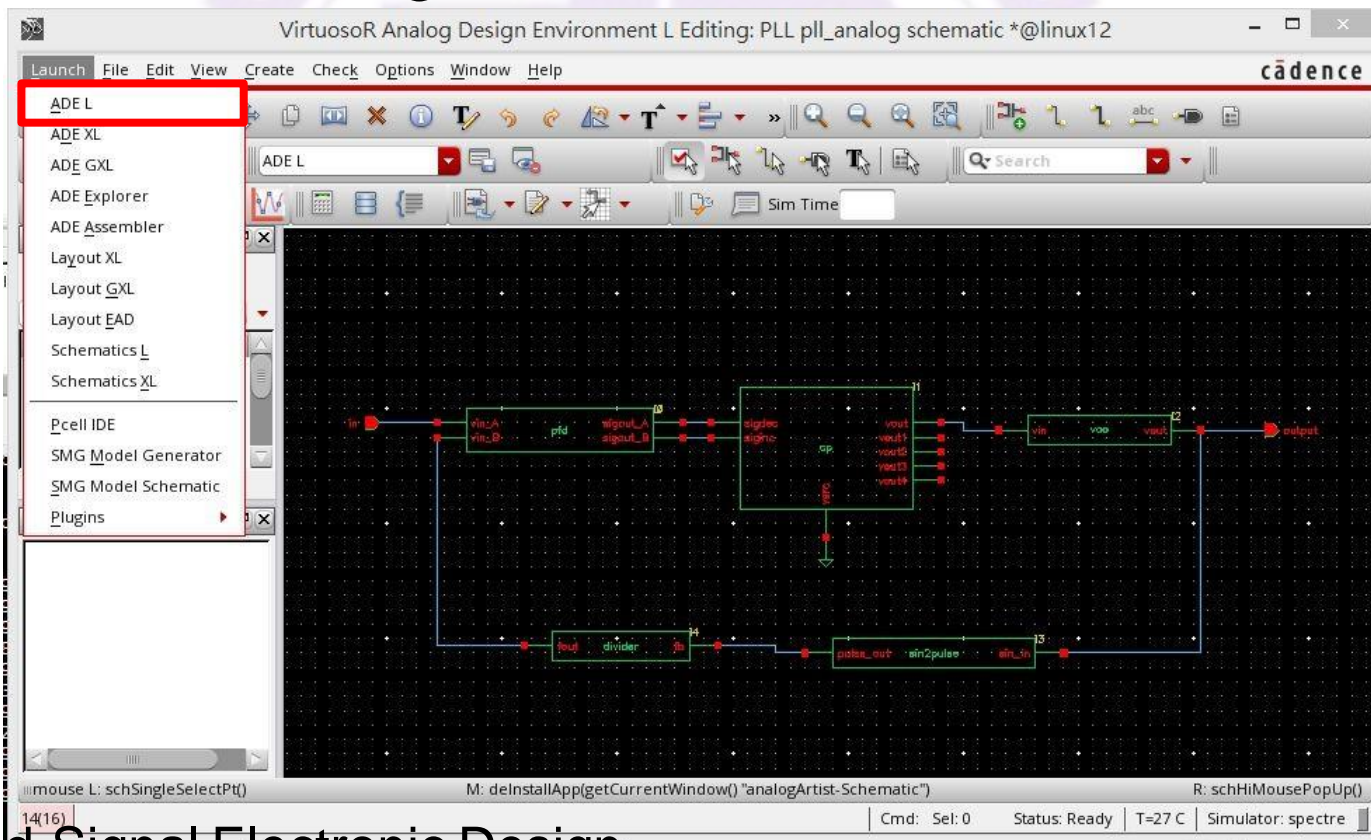


Add connection: Create → wire (w) gnd in the “basic” library  
Add input/output: Create → pin (p)



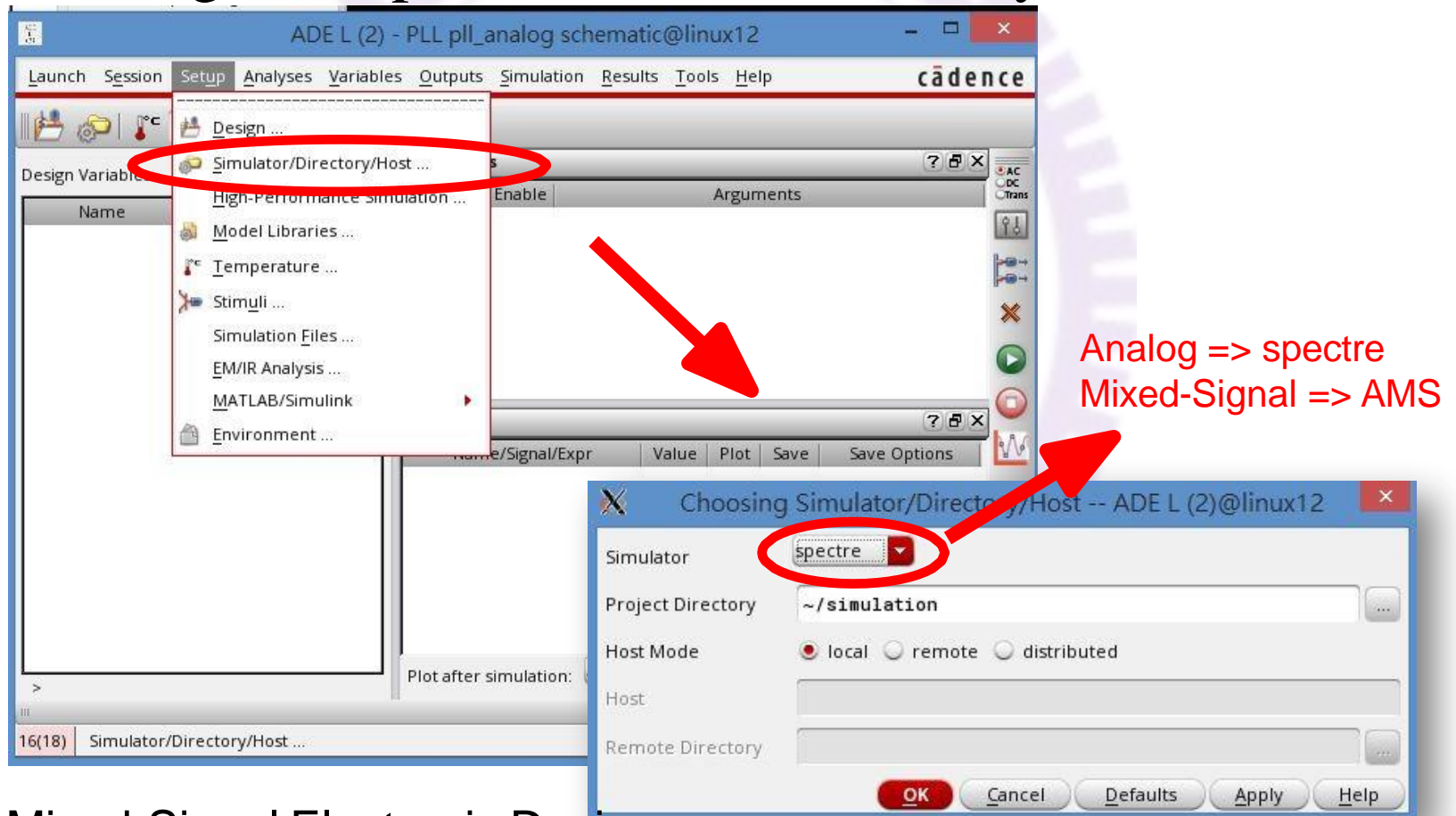
# Simulation Environment

- Open Analog Design Environment (ADE) in schematic editing window





- Through Setup -> Simulator/Directory/Host

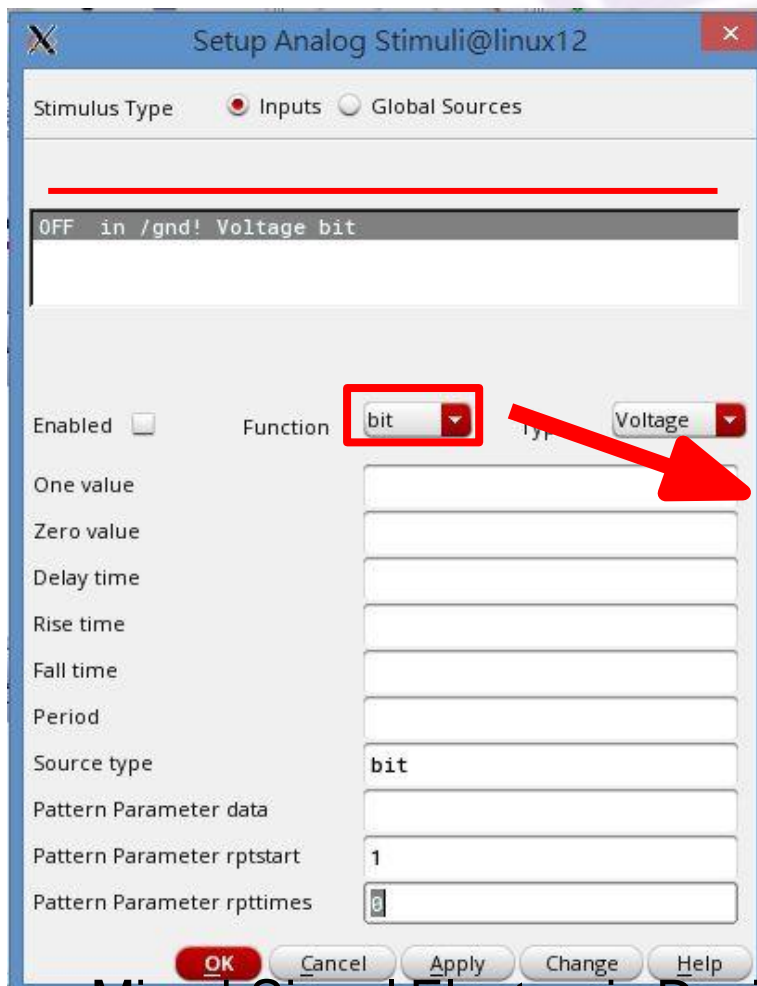


# Give input information(1/2)

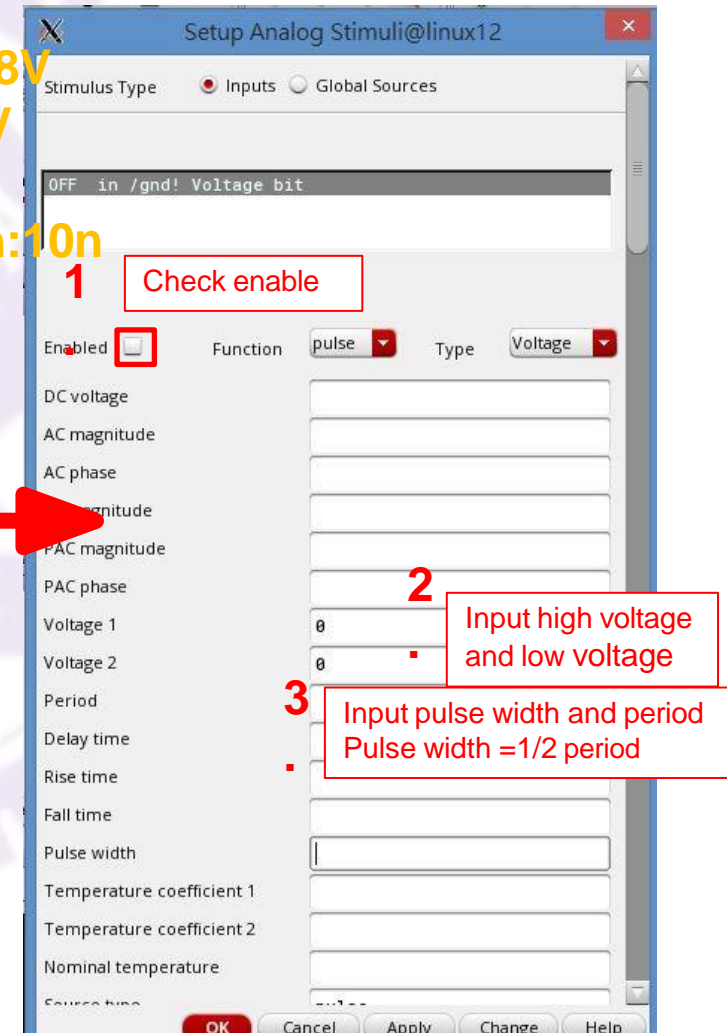
The screenshot shows the Cadence ADE L (2) - PLL pll\_analog schematic@linux12 interface. The 'Setup' menu is open, and the 'Stimuli ...' option is highlighted with a red box. A red arrow points from this option to the 'Setup Analog Stimuli@linux12' dialog box. In the dialog, the 'Stimulus Type' is set to 'Inputs'. The 'Stimulus' field is highlighted with a red box and contains the text '0FF in /gnd! Voltage bit'. A red arrow points from this field to the text 'Input pin name in schematic'. The 'Function' is set to 'bit' and the 'Type' is set to 'Voltage'. The 'Enabled' checkbox is unchecked. The 'One value', 'Zero value', 'Delay time', 'Rise time', 'Fall time', and 'Period' fields are empty. The 'Source type' is set to 'bit'. The 'Pattern Parameter data' field is empty. The 'Pattern Parameter rptstart' is set to '1' and the 'Pattern Parameter rpttimes' is set to '0'. The 'OK', 'Cancel', 'Apply', 'Change', and 'Help' buttons are at the bottom.

If choose spectre as simulator, there only has stimuli

# Give input information(2/2)



Voltage1:1.8V  
Voltage2:0V  
Period:20n  
Pulse width:10n



# Choose Analysis Type

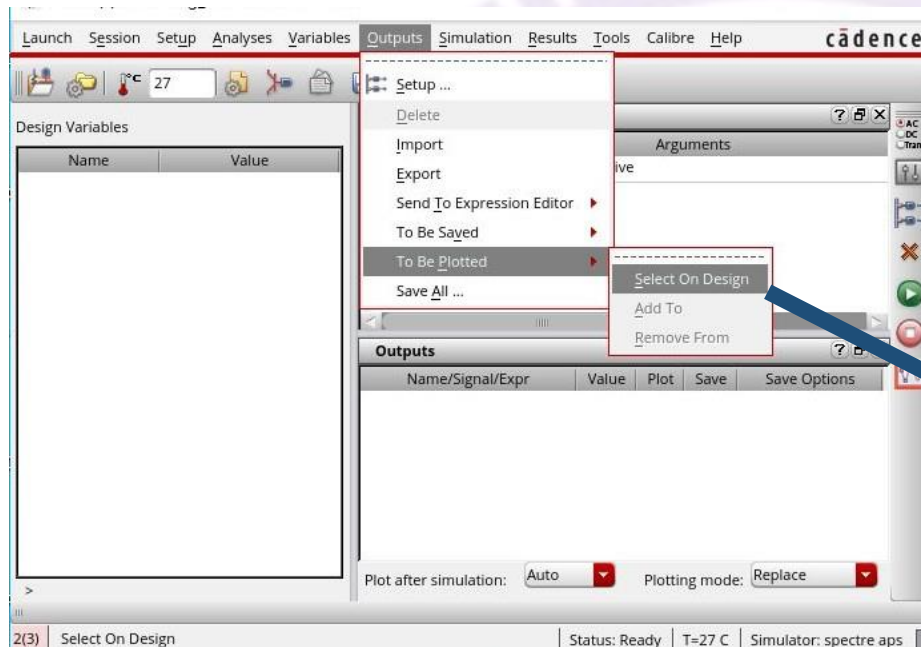
The screenshot shows the Cadence ADE L (2) - PLL\_pll\_analog schematic@linux12 window. The 'Analyses' tab is active, and a 'Choosing Analyses' dialog box is open. The dialog box has a red border and contains the following elements:

- Analysis Selection:** A grid of radio buttons for various analysis types. The 'tran' (Transient) analysis is selected.
- Transient Analysis Section:**
  - Stop Time:** A text box containing '6u'.
  - Accuracy Defaults (errpreset):** Three checkboxes: 'conservative' (checked), 'moderate', and 'liberal'.
  - Transient Noise:** A checkbox that is currently unchecked.
  - Dynamic Parameter:** A checkbox that is currently unchecked.
  - Enabled:** A checkbox that is checked, highlighted with a red box.
- Buttons:** 'OK', 'Cancel', 'Defaults', 'Apply', and 'Help' buttons at the bottom.

Red annotations on the right side of the image provide instructions:

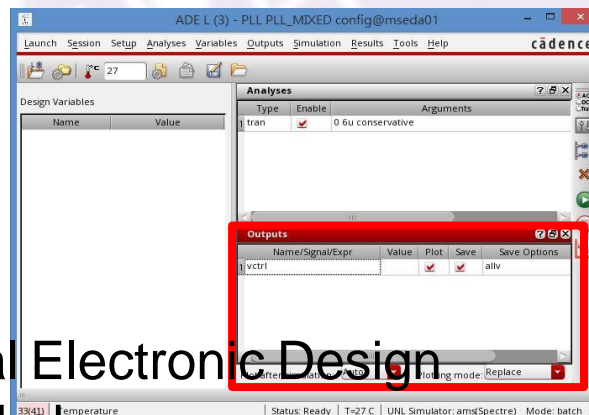
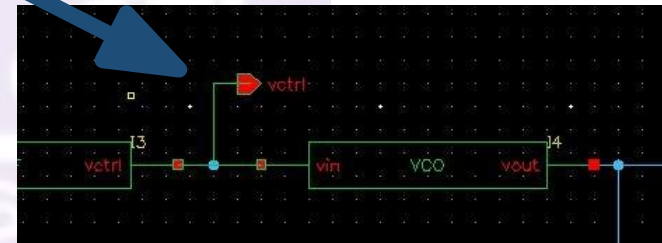
- Choose tran analysis** (pointing to the 'tran' radio button)
- Set the simulation time and the accuracy flag** (pointing to the 'Stop Time' and 'Accuracy Defaults' section)
- Check enable to simulation** (pointing to the 'Enabled' checkbox)

# Save Output Nodes



To Be Plotted:

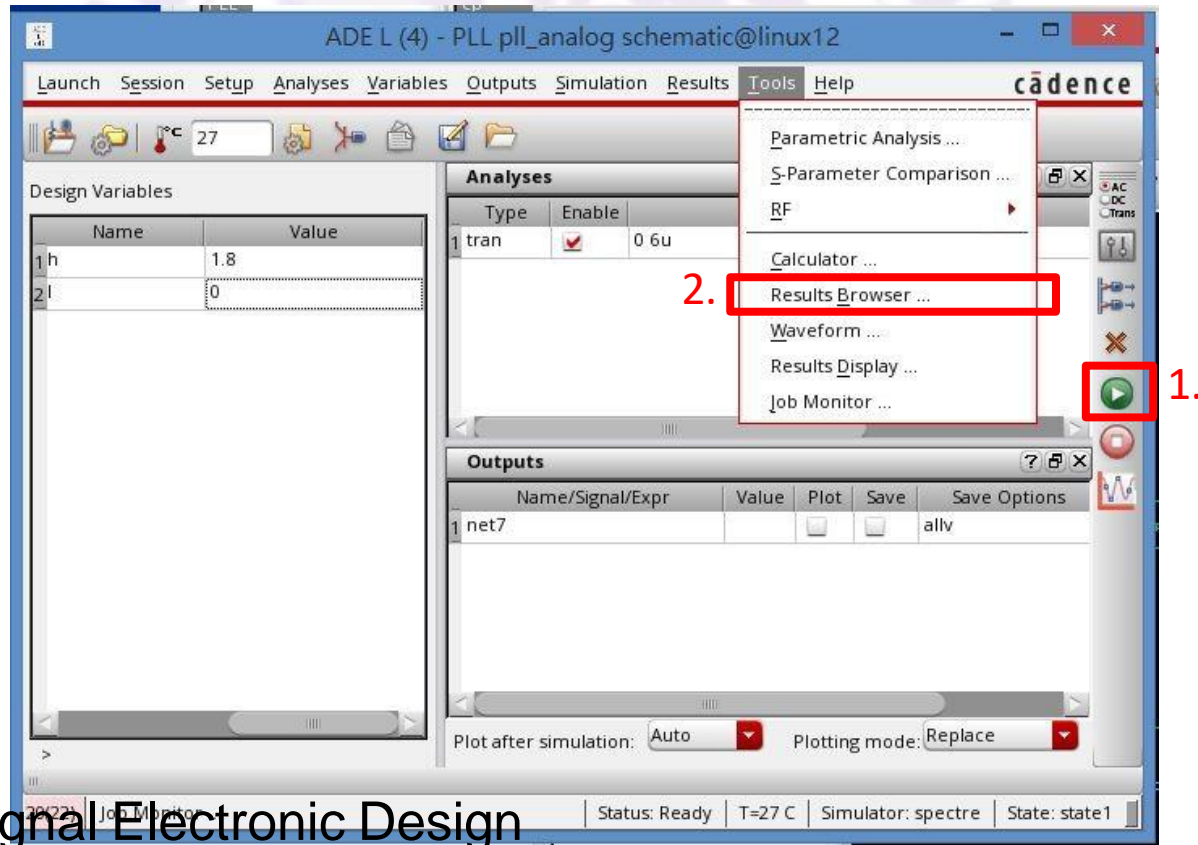
- ☐ Select on **line** or **node**
- ☐ Press **Esc** to cancel





# Submit the Simulation

- Execute the simulation job with Run
- Tools → Results Brower



# Outline

---

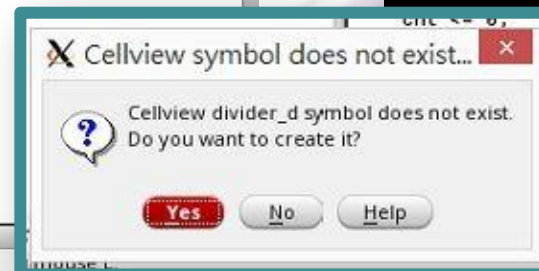
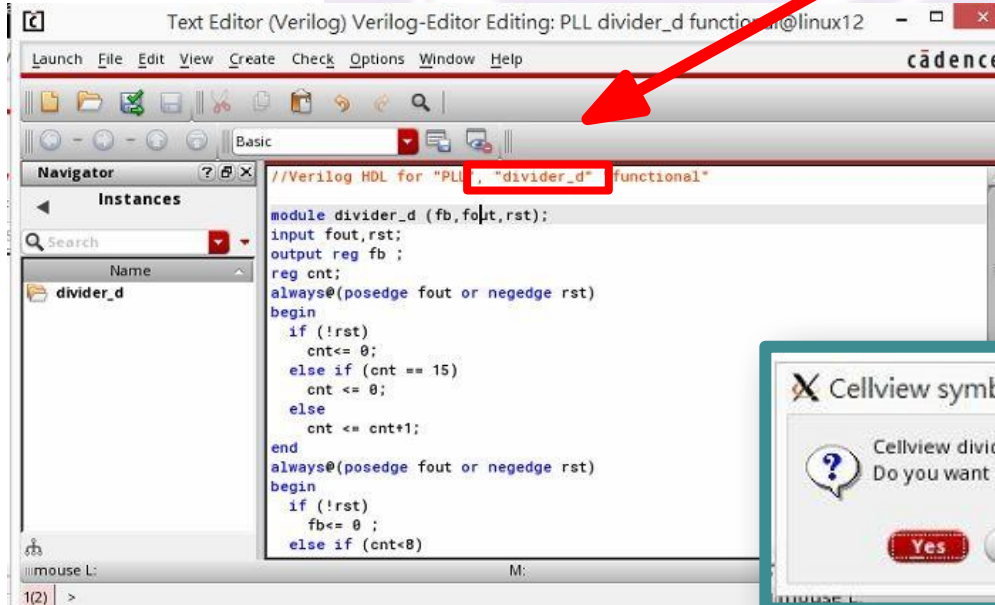
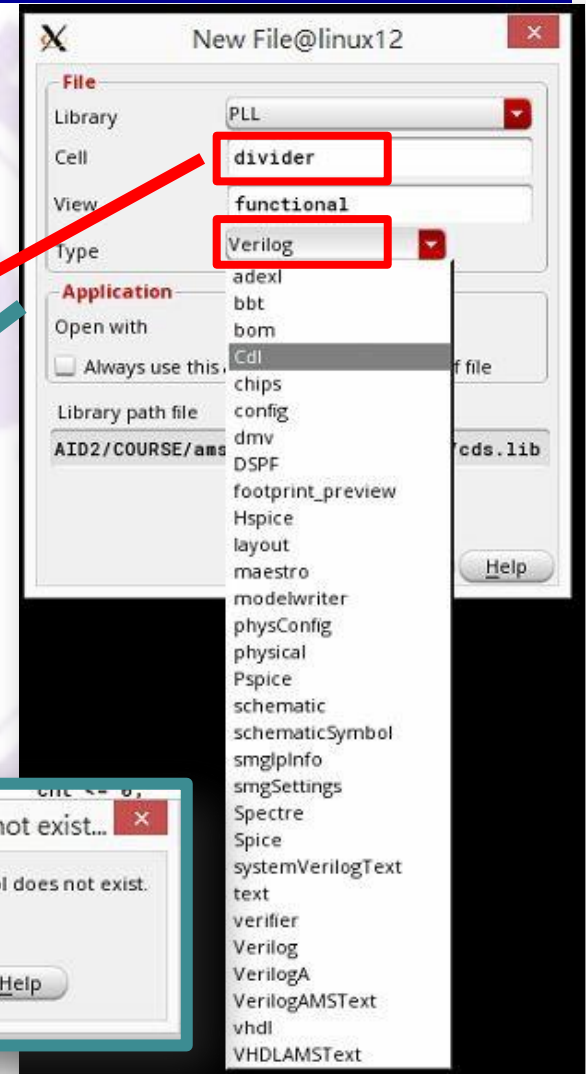
- Introduction to AMS
- **AMS Simulation Setup**
  - Analog Simulation with Verilog-A Model
  - **Mixed-Signal Behavioral Model Simulation**
    - Lab1: Analog Model Simulation with AMS
    - Lab2: Mixed-Signal Model Simulation with AMS
    - Lab3: 4-bit ADC with Verilog-AMS



# Create Verilog Cells

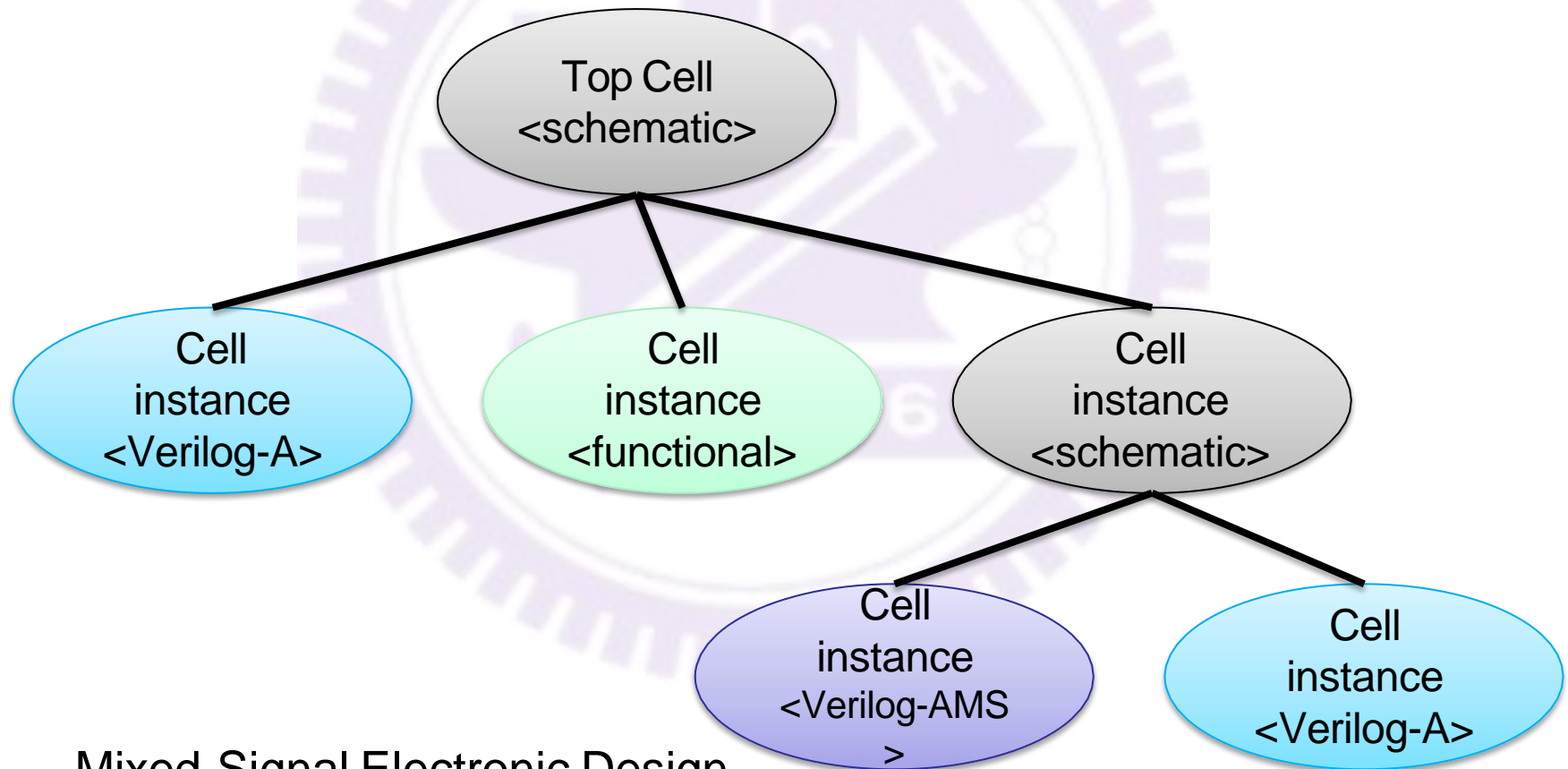
- Create library
- Create cell view & symbol
  - Verilog-A models
  - Verilog models
  - Schematic
  - Config
- Set ADE

The cell name  
must be the same



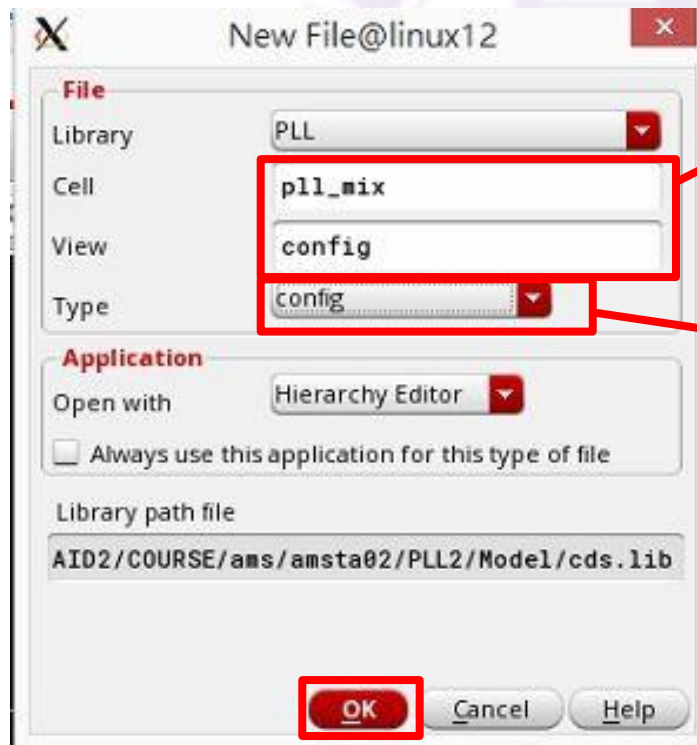
# Design Hierarchy – AMS

- Before creating the top schematic cell (add instance and connection), creating a config view for AMS simulation



# Create Config View for Simulation

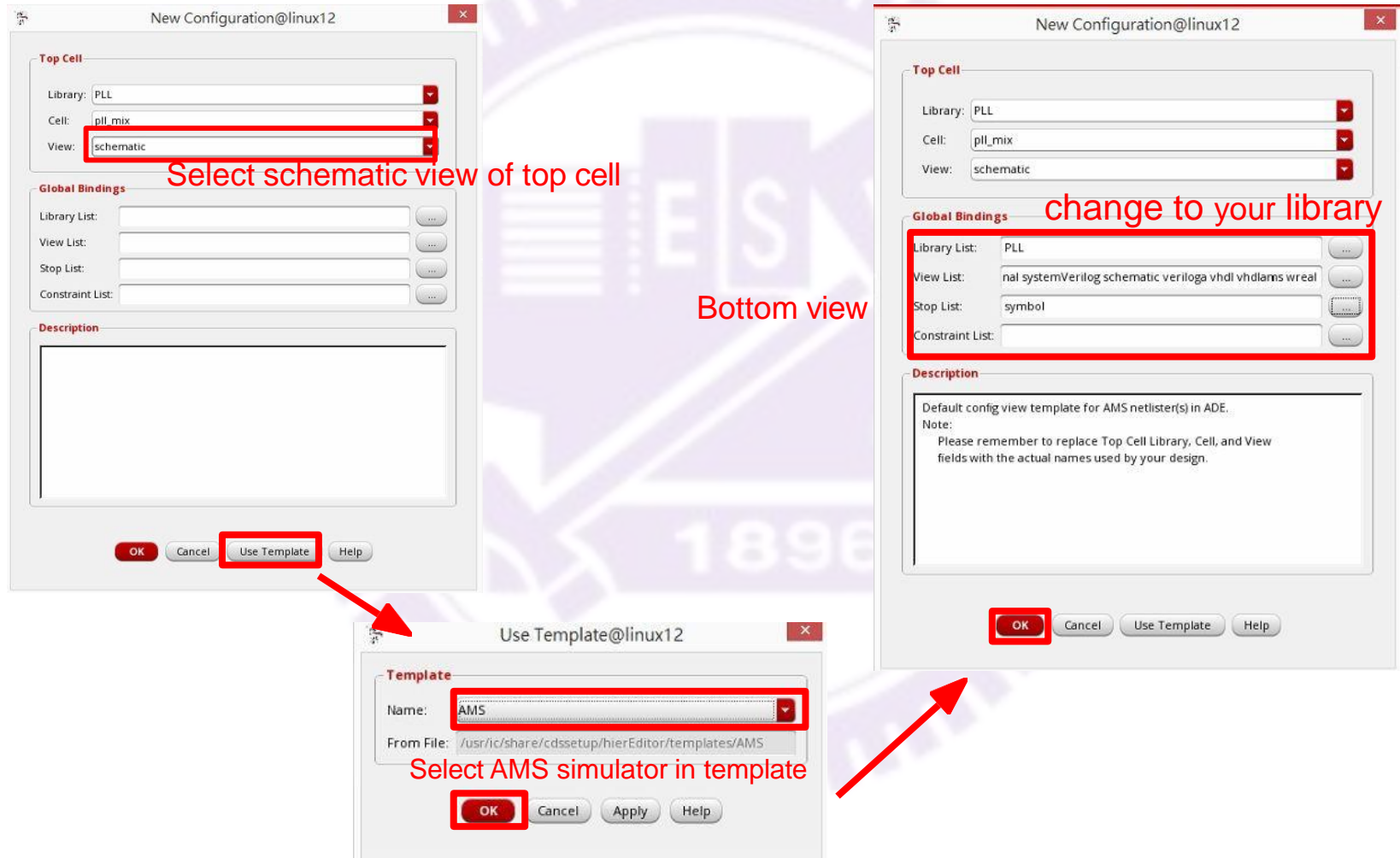
- The mixed-signal simulation hierarchy is controlled by **Hierarchy-Editor**
  - It must have to be defined in the **config** cell view.



Cell name is top schematic cell for simulation

Choose **config** type

# Set New Configuration



# Configuration Setting

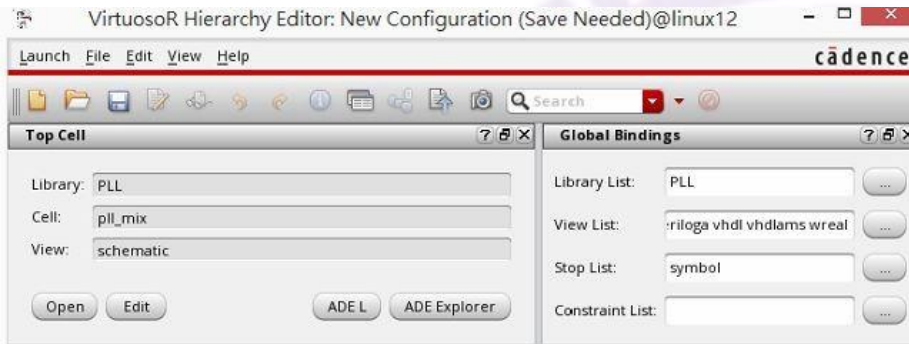


Table View Tree View

Cell Bindings

Library	Cell	View Found	View To Use	Inherited View List
PLL	cp	veriloga		spectre spice pspice...
PLL	divider_d	functional		spectre spice pspice...
PLL	lf	veriloga		spectre spice pspice...
PLL	pfd	veriloga		spectre spice pspice...
PLL	pll_mix	schematic		spectre spice pspice...
PLL	sin2pulse	veriloga		spectre spice pspice...
PLL	vco	veriloga		spectre spice pspice...

The hierarchy-editor will list all cells and views in the cell bindings

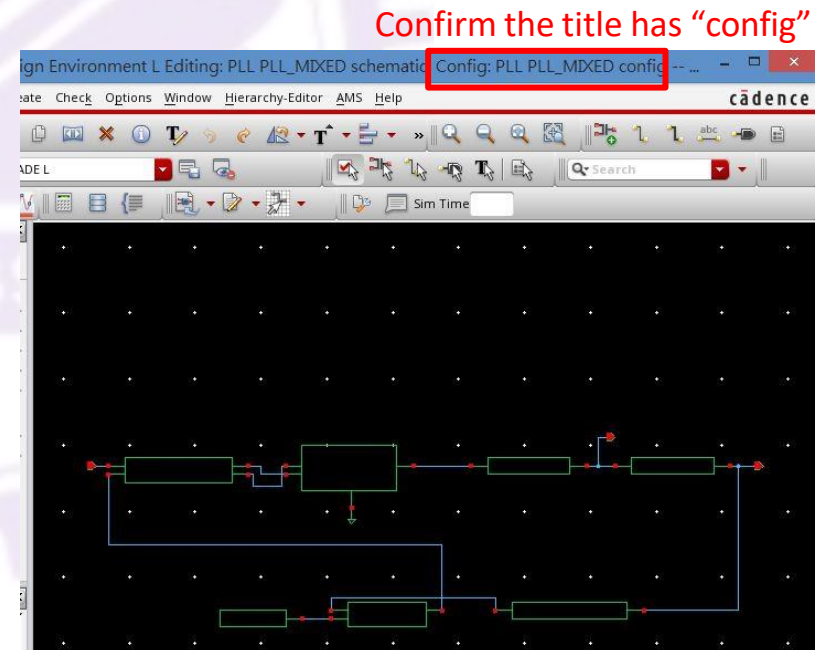
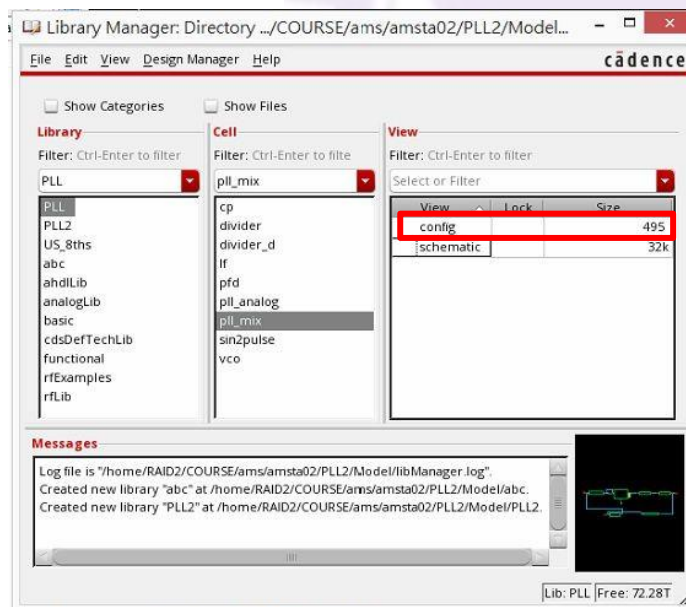
Must to save and recompute!





# Open Simulation Tool

- Finish create config
  - Click config at library manager to open simulation tool
  - The simulation steps are the same as analog
    - Except give input information



# Digital Stimulus

- Create a behavioral or functional view for the stimulus block
  - The stimulus (Verilog) could be created to symbol view as the same procedure with digital cell

```
//Verilog HDL for "PLL", "stimulus_D" "functional"

`timescale 1ns/10ps
module stimulus_D (rst);
output rst;
reg rst;
initial begin
    rst=1'b0;
    #1 rst=1'b1;
end
endmodule
```



# Analog Stimulus

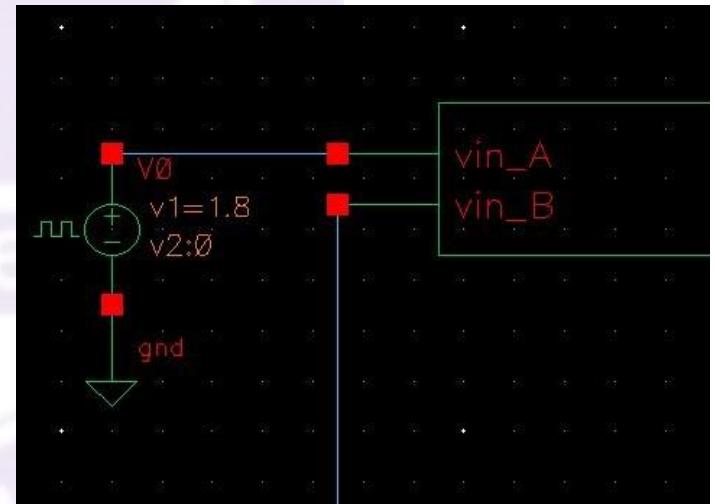
- The analog stimulus can be added as circuit instance

## Add Instance

Choose  
analogLib  
& vpulse cell

Set  
voltage &  
frequency

Library: analogLib  
Cell: vpulse  
View: symbol  
Add Wire Stubs at: all terminals  
Array: Rows 1, Columns 1  
Frequency name for 1/period:   
Noise file name:   
Number of noise/freq pairs: 0  
DC voltage:   
AC magnitude:   
AC phase:   
XF magnitude:   
PAC magnitude:   
PAC phase:   
Voltage 1: 0 V  
Voltage 2: 0 V  
Period:



# 1. Setup

- ✓ Simulator choose AMS

# 2. Analysis

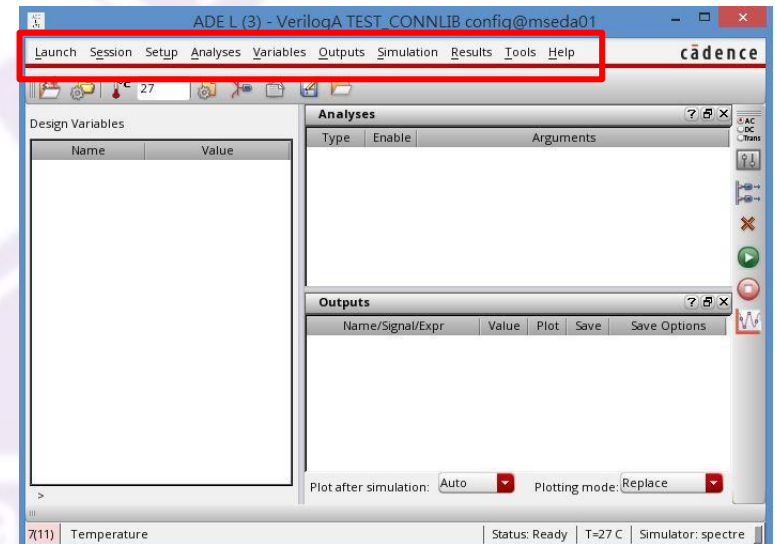
- ✓ Tran analysis
- ✓ Set simulation time and enable

# 3. Outputs

- ✓ Save all or select on design

# 4. Run

# 5. Waveform viewer



# Outline

---

- Introduction to AMS
- **AMS Simulation Setup**
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - **Lab1: Analog Model Simulation with AMS**
    - Lab2: Mixed-Signal Model Simulation with AMS
    - Lab3: 4-bit ADC with Verilog-AMS

- All Verilog-A models are given
  - PFD, CP, LF, VCO, sin2plus, divider
- Import the Verilog-A models and setup ADE



# Additional Description

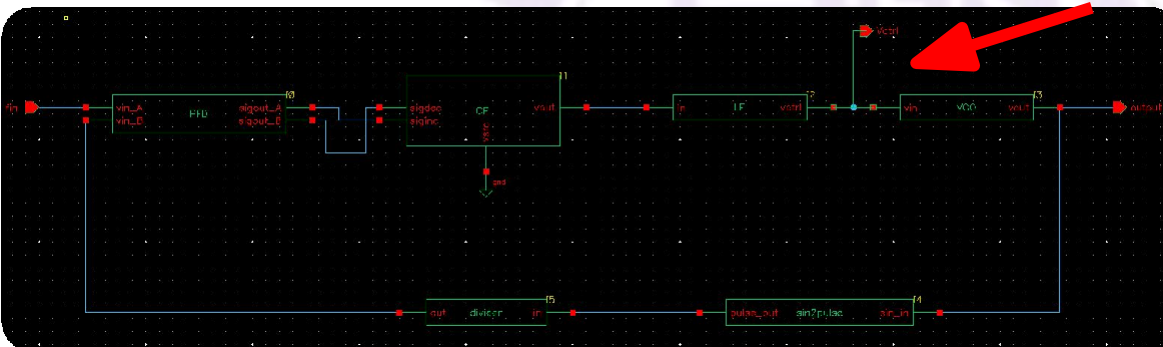
- Connection
  - Connect up (*sigout\_A*) signal of PFD to *siginc* signal of CP
  - Connect dn (*sigout\_B*) signal of PFD to *sigdec* signal of CP
  - Connect *in* signal of *divider* to *sin2pulse*
  - Charge pump(CP)
    - Connect vsrc to GND
- $VDD = 1.8V$     $GND = 0V$
- Simulator: **spectre**
- Input
  - **$F_{in}=50M$  Hz : Set the ADE Stimuli**
- Simulation time  $\geq 6\mu s$ 
  - Related to the lock time of PLL

# Results

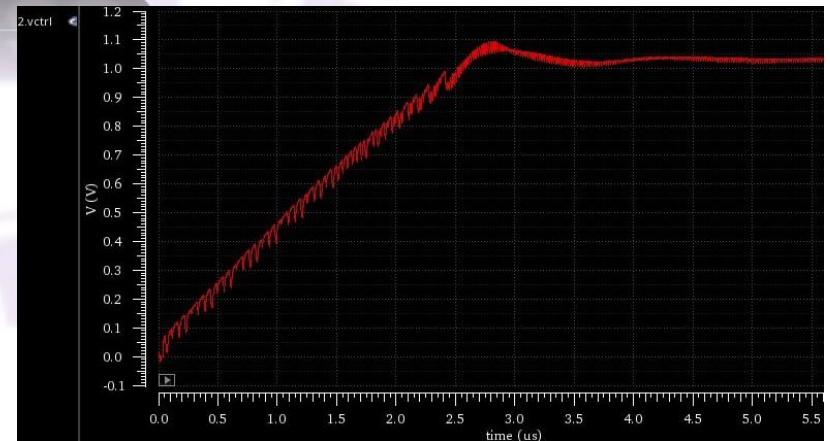
- Show the waveform of vctrl signal and schematic view

Top schematic cell

Observe the waveform of Vctrl



Open Results Brower





# Outline

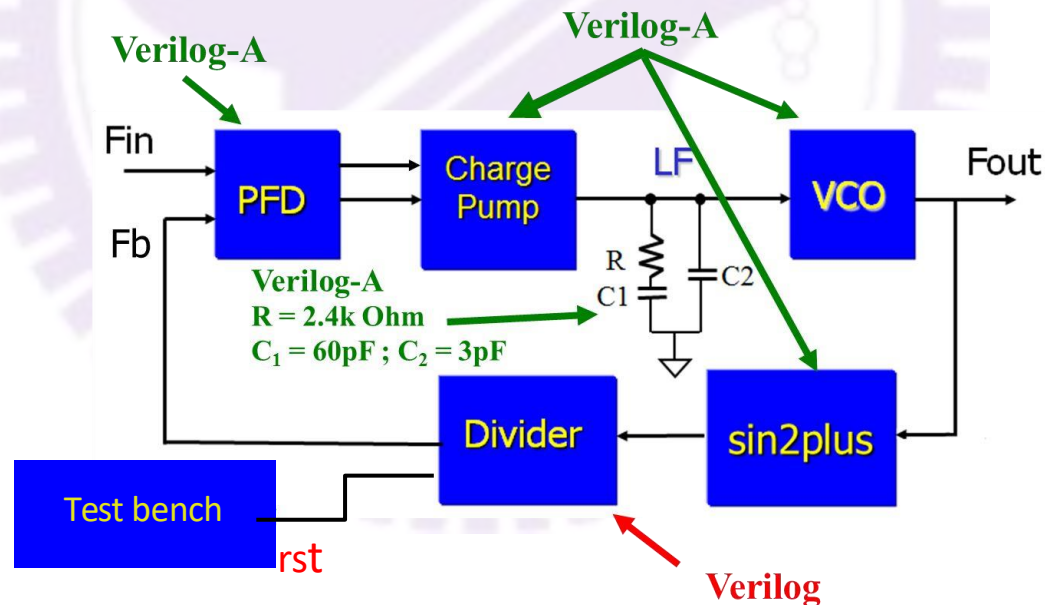
---

- Introduction to AMS
- **AMS Simulation Setup**
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - Lab1: Analog Model Simulation with AMS
    - **Lab2: Mixed-Signal Model Simulation with AMS**
    - Lab3: 4-bit ADC with Verilog-AMS



# PLL Mixed-Signal Model

- All models are given
  - Analog model (Verilog-A): PFD, CP, LF, VCO, sin2plus
  - Digital model (Verilog): divider
- Import all models and create a testbench
  - Testbench : generate **rst** signal for divider



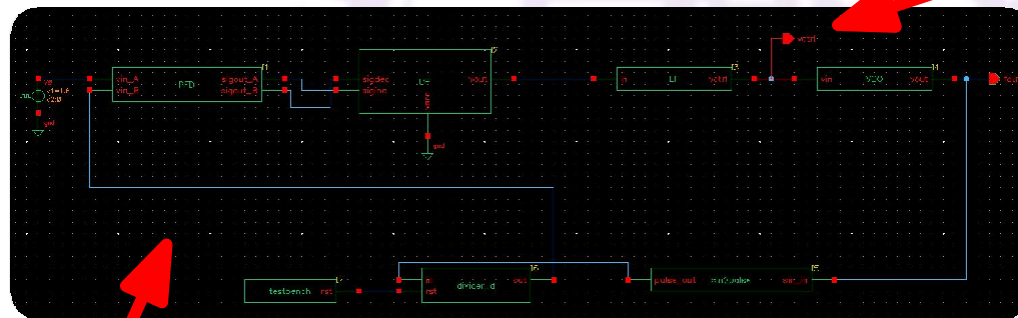
# Additional Description

- Connection
  - Connect up (*sigout\_A*) signal of PFD to *siginc* signal of CP
  - Connect dn (*sigout\_B*) signal of PFD to *sigdec* signal of CP
  - Connect *in* signal of *divider* to *sin2pulse*
  - Charge pump(CP)
    - Connect vsrc to GND
- $VDD = 1.8V$     $GND = 0V$
- Simulator: **AMS**
- Input
  - **$F_{in}=50M$  Hz, Adding voltage source instance**
- Simulation time  $\geq 6\mu s$ 
  - Related to the lock time of PLL

# Results

- Show the waveform of vctrl, output of the divider and schematic view

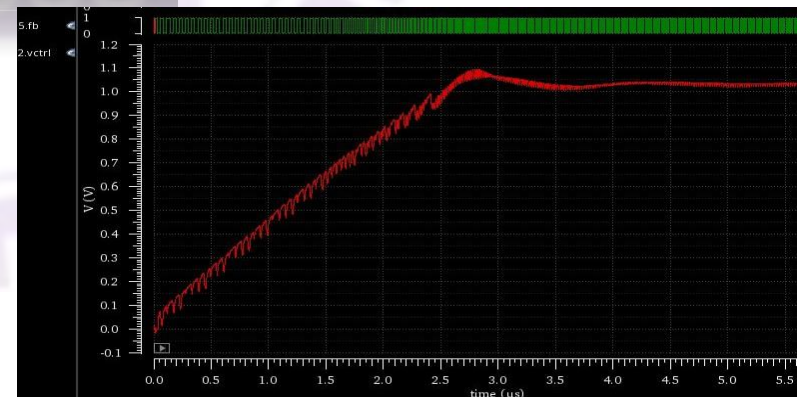
Top schematic cellview



Observe the waveform of Vctrl

Observe the output of divider.v

Open Results Brower

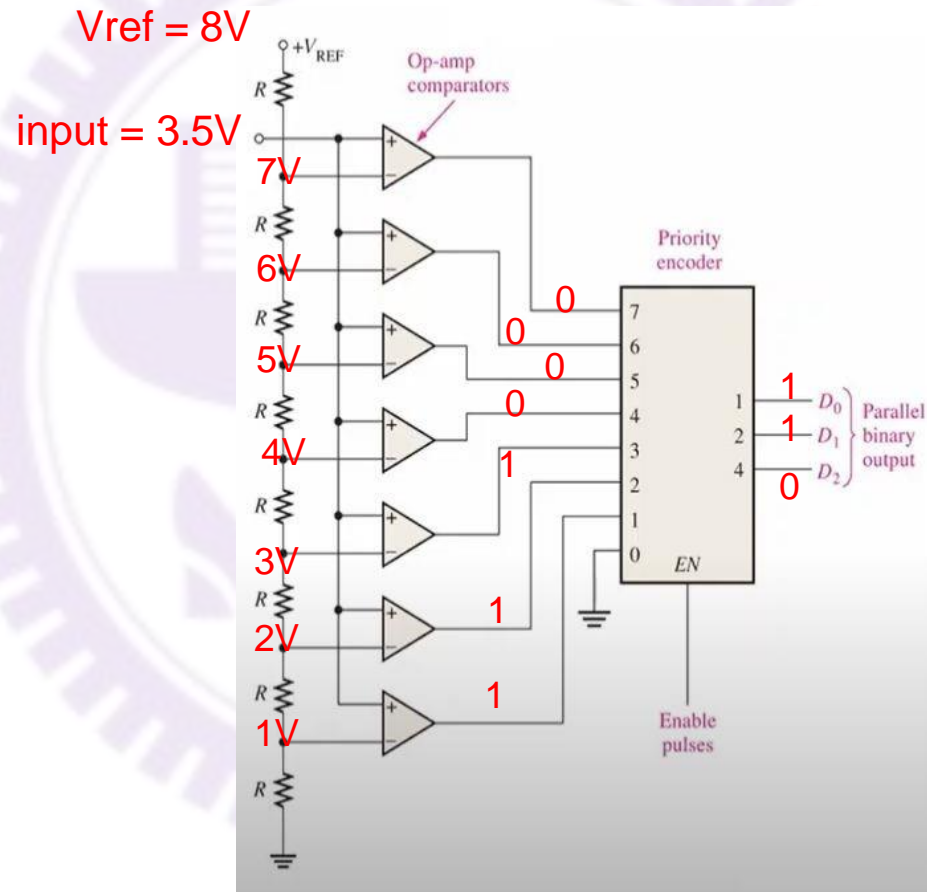


# Outline

---

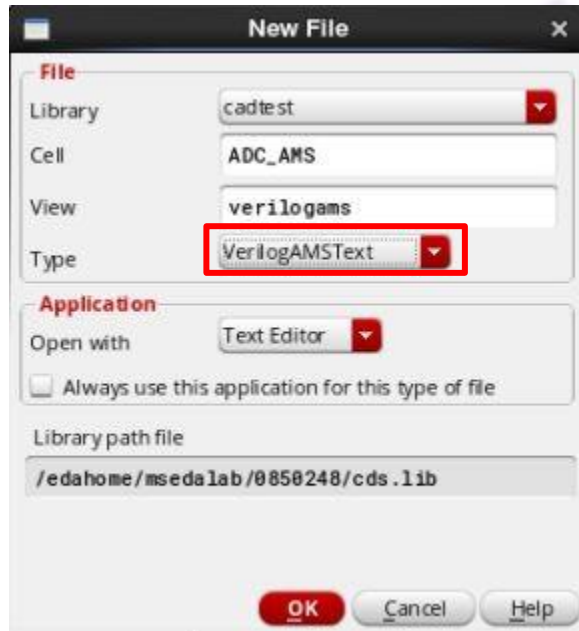
- Introduction to AMS
- **AMS Simulation Setup**
  - Analog Simulation with Verilog-A Model
  - Mixed-Signal Behavioral Model Simulation
    - Lab1: Analog Model Simulation with AMS
    - Lab2: Mixed-Signal Model Simulation with AMS
    - **Lab3: 4-bit ADC with Verilog-AMS**

# Description





# Type : AMS



```
connectmodule a2d(i,o);  
  parameter vdd = 1.0;  
  ddiscrete o;  
  input i;  
  output o;  
  reg o;  
  electrical i;  
  always begin @(cross(V(i) - vdd/2,+1))o = 1; end  
  always begin @(cross(V(i) - vdd/2,-1))o = 0; end  
endmodule
```

Support output reg

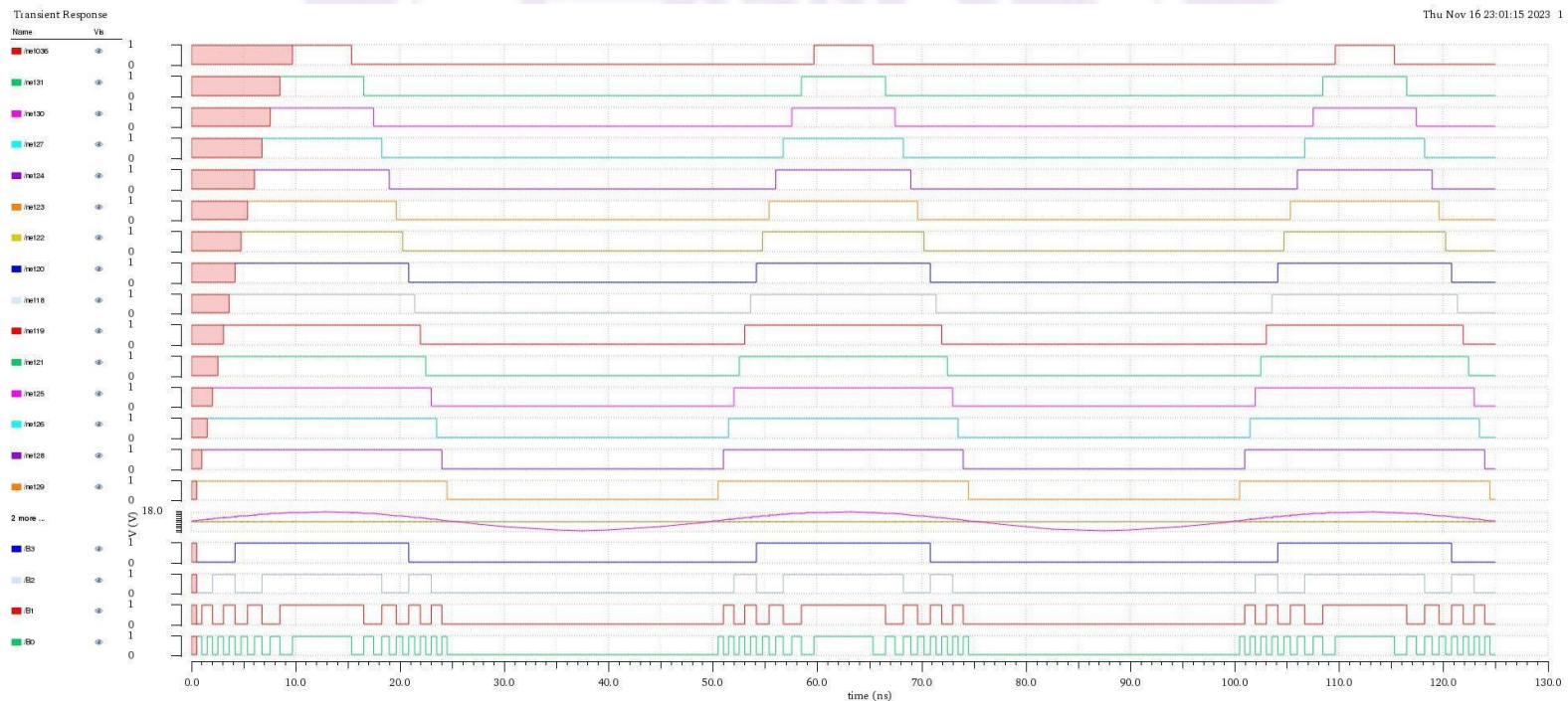
# Parameters of ADC Model

- Input : sin wave
- Amplitude : 16V, Frequency : 20M
- Output : 4 bit output
- $R=0.5k$ ,  $V_{ref}=16V$
- Analysis Type : tran, 125n
- Simulator: **AMS**

B3,B2,B1,B0 (bit)	Input (V)
0000	0
0001	1V
0010	2V
0011	3V
0100	4V
0101	5V
0110	6V
0111	7V
1000	8V
1001	9V
1010	10V
1011	11V
1100	12V
1101	13V
1110	14V
1111	15V

# Results

- Hand in:
  - The waveform of outputs of comparators, encoder and input
  - Schematic cellview



# Hand in

---

- Please upload a compressed file includes:
  - **Programming files** (Verilog and Verilog-A files)
    - Lab2: Stimuli (testbench.v)
    - Lab3: Encoder file(.vams or .v), comparator file(.vams or .va)
  - **Mini report**
    - Three simulation waveforms
      - Lab1: Analog simulation (Vctrl)
      - Lab2: Mixed-signal simulation (Vctrl & output of the divider)
      - Lab3: ADC (outputs of comparators, encoder and input)
    - Three schematic cellviews
    - What you have learned from this homework
    - Questions and solutions
- **Deadline: 23:55, December 31, 2023**