Passing CDF parameters from instantiated Symbol to Schematic

Tags: cadence, schematic

■ From blog.intgckts.com/author/sameeksha/

In Cadence, we can pass parameters individually from each instantiated symbol to schematic using Component Description Format(CDF) parameters.

Where do I require?

- Suppose you want to test your design idea, where you require two op-amps with different specifications but want to use same macro model/schematic for both opamps.
- Suppose you want to create buffer chain with a series of inverters. In each stage you want to scale the size of inverter by "X" compared to preceding stage. Again you want to use same inverter model across the entire chain.
- many more….

How to?

We can use of **parent parameters (pPar)** and **instance parameters (iPar)** to pass parameters from instantiated symbol to schematic. The following CMOS inverter example explain this.

Consider an inverter that contains a NMOS and PMOS transistors. We want to pass width and length of mos transistors as parameters to the schematic from the symbol where it is instantiated.

So open the schematic of the inverter, edit the properties of NMOS and PMOS transistors with the parameters as shown below,
Length: pPar("lp") and Width: pPar("wp") for PMOS,
Length: pPar("ln") and Width: pPar("wn") for NMOS.

CDF Parameter	Value	Display
Model name		off
Multiplier	1	off
Width	pPar("wn") M	off
Length	pPar("ln") M	off

Now save the inverter schematic and create a symbol for the schematic through the menu:

Design >> Create Cellview >> From Cellview

Once the inverter symbol is created add label as follows to show the parameters in symbol view when it is instantiated.
Label: [@wp:%:]/[@lp:/%:]
Label: [@wn:%:]/[@ln:/%:]

Save the edits, and close the symbol editor.

Select the CIW of icfb, go to: **Tools >> CDF >> Edit**
Now select **CDF Type : Base**, click on **Browse** button and point to your Library Name/Cell Name.

Click on **Add** button in **Component Parameters** section to open **Add CDF Parameter** form. Add the parameters as shown below.

Then click on **Apply** button, repeat the same process for other parameters (lp,wn and ln) also and finally click OK. Now your **Component parameters** should like shown below(of-course the values may be different).

You are done with assigning global parameters for the whole library. Now you can individually assign the values to all parameters in each instantiation.

A test bench schematic with two inverters instantiated is shown below. The first inverter is sized to 2u/1u and second is sized to 6u/3u, scaled by 3X compared to first, using CDF parameters.
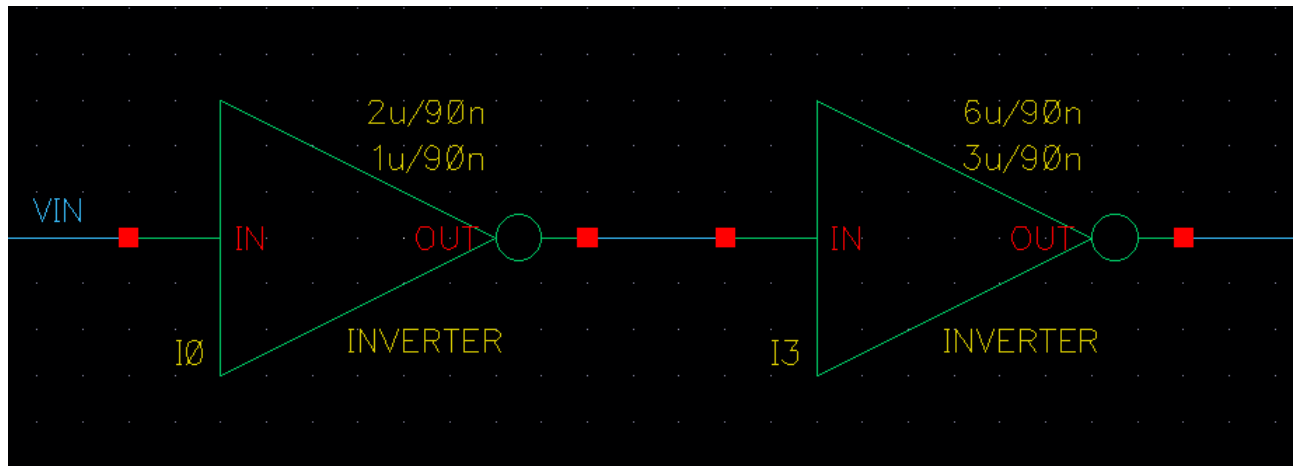
iPar()

To inherit the parameters form the instance parameters itself then use ipar().

Example : We can define finger width in terms of finger length in a transistor
Finger Width : *5*iPar("l")*