



Schematics and Symbols

Prof. Dr. P. Fischer

Lehrstuhl für Schaltungstechnik und Simulation
Uni Heidelberg



What are Symbols ?


- Very often, a circuit (schematic) can be re-used.
- Instead of copying everything, we can 'include' the schematic into another schematic
- In order to identify the nets, we need a **symbol**
 - This is a new **view** type

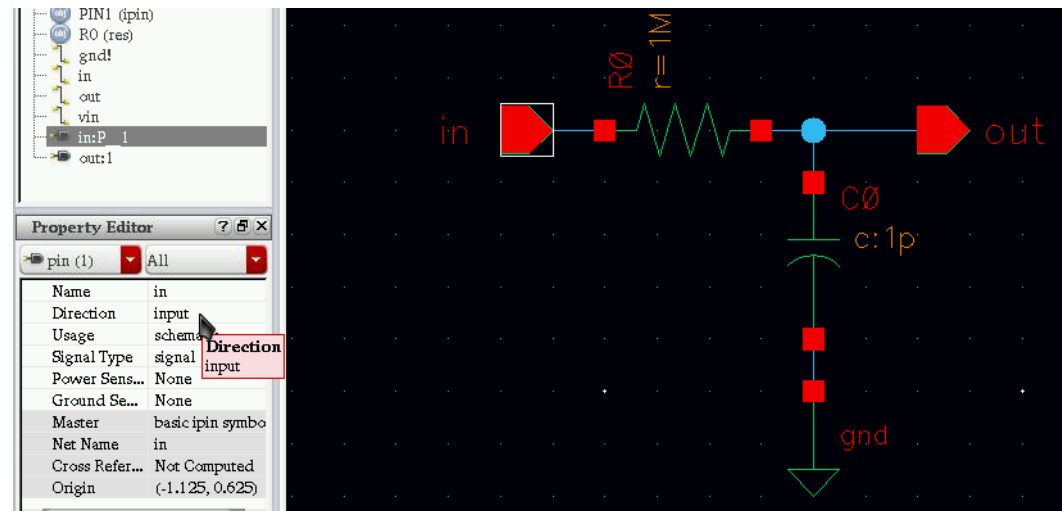
| View | Lock | Size |
|-----------|------|------|
| layout | | 17k |
| schematic | | 30k |
| symbol | | 23k |

- The nets which are passed to the outside world must be connected to **pins** in the schematic.
 - For **each pin in the schematic** we also need a **pin in the symbol**.
- Pins must have the **same name** as the connected net
- They can be **Input / Output / inputOutput** (see later)



Preparing the Schematic

- The easiest way to create a symbol starts from a schematic
- Using **Create** → **Pin** (Ctrl-P or button ) , create pins for all signals that should be visible 'outside'
 - **outputs** are signals that will drive to other cells
 - **inputs** only receive signals. They **must** be connected later
 - **InputOutput** are most general. Only use if you have to!

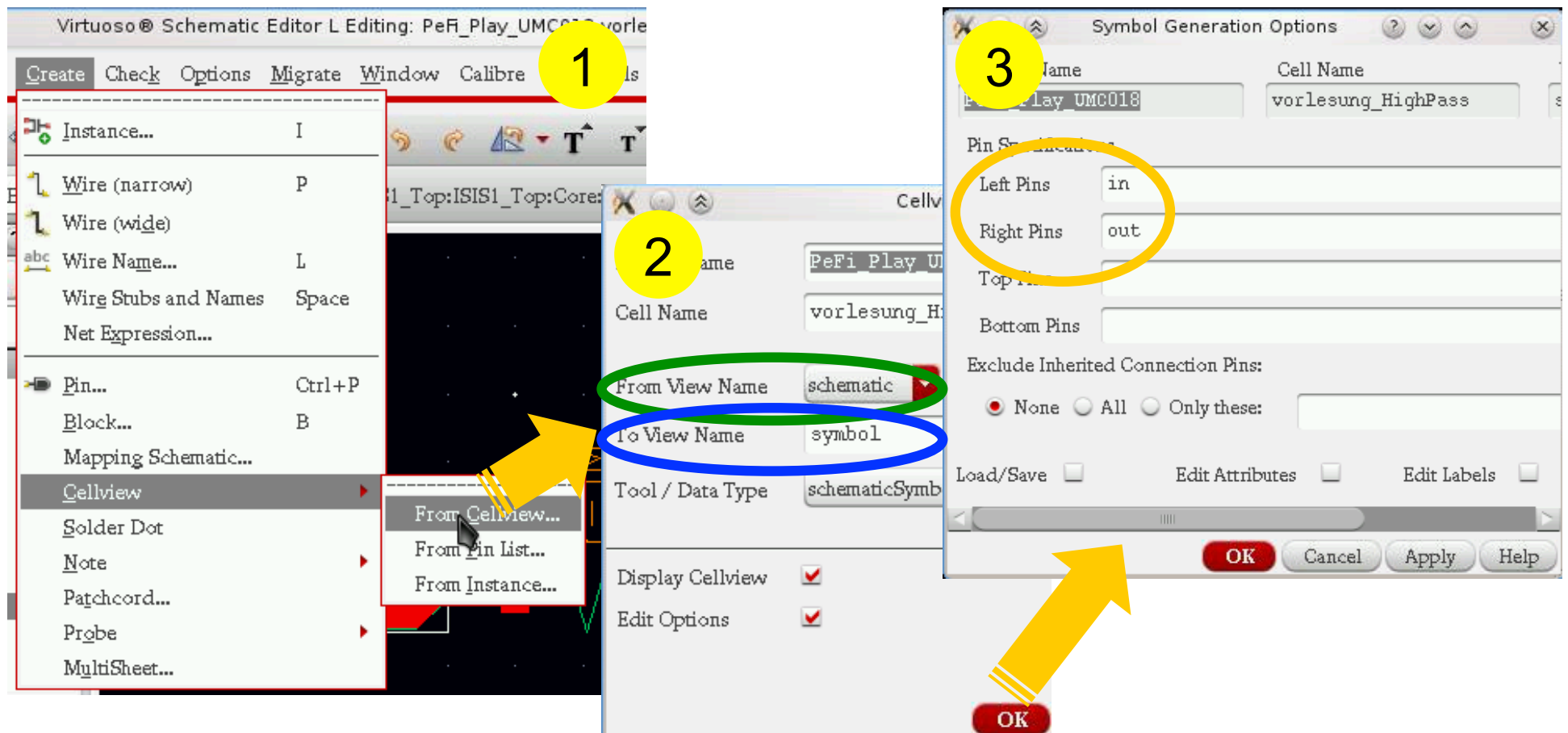


- A pin **labels** the net, i.e. a further label is not required
- Better remove all symbols used for simulation (sources..)



Creating a Symbol from the Schematic

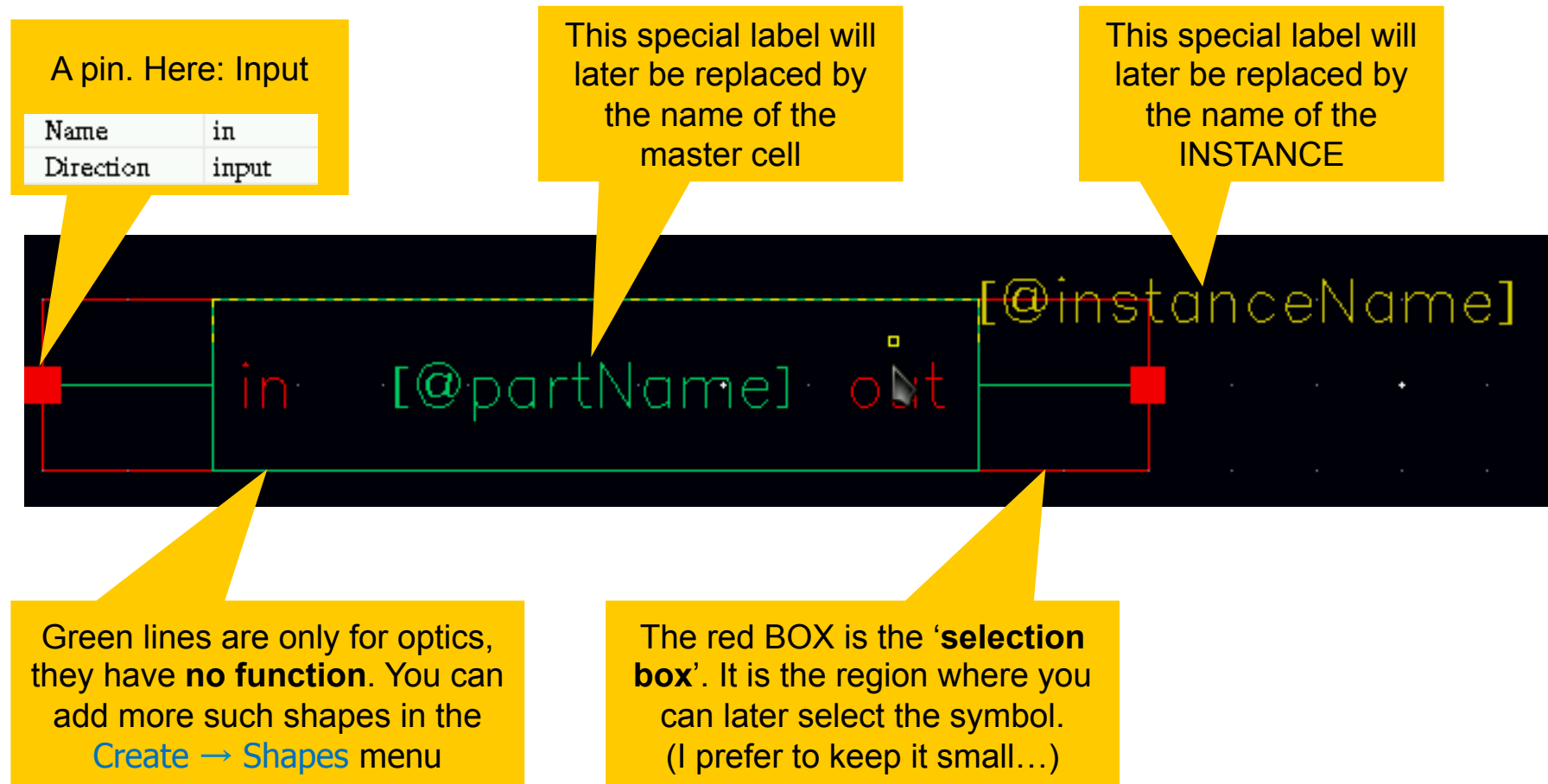
1. Select Create → Cellview → From Cellview
2. Check that 'From View' is *schematic* and 'To view' is *symbol*
3. Press ok. In the next window, select the pin locations





Editing the Symbol

- A symbol template is created:

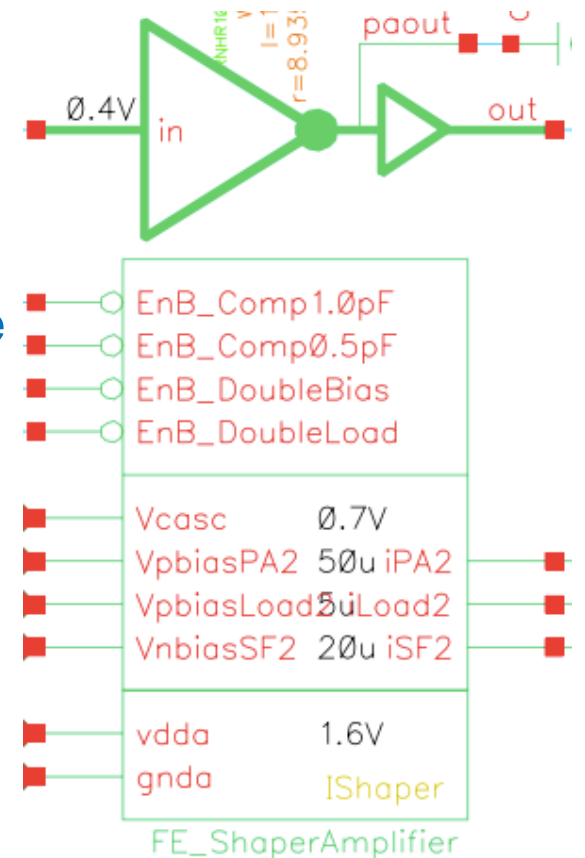
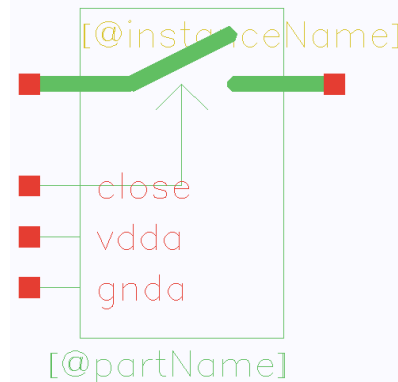


- You can set the origin under [Edit → Origin](#)



Make Nice Symbols!

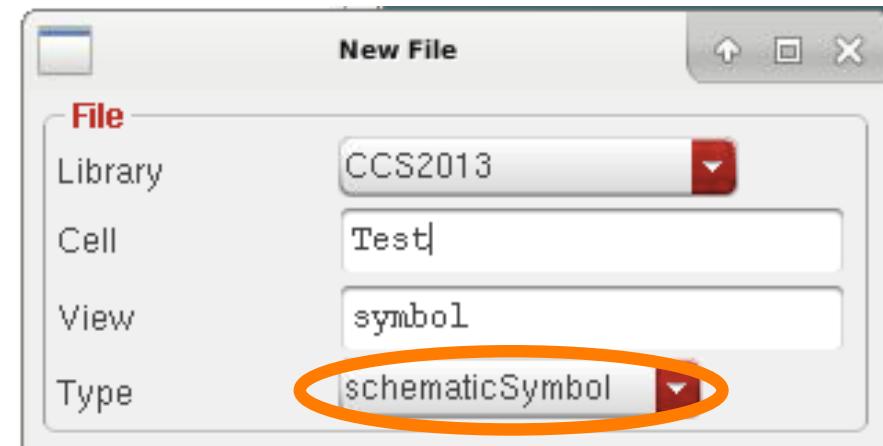
- Your schematics get more readable if the symbols are 'nice':
 - Power (if present at pins) may be grouped at the bottom
 - Group bias signals, use 'good' names
 - Inputs are left / outputs are right
 - Digital signals are grouped
 - Active Low signals have a bullet
 - Clocks have with a triangle
 - Add a little drawing of the functionality
Create→Note→Shape or Create→Shape
 - Add text: Create→Note→Text
 - You may delete trivial labels





Creating a Symbol from Scratch

- You can also create an (empty) new symbol directly from the library browser with **File** → **New** → **Cell View...** with view type ***schematicSymbol***



- You must then place all pins, boxes, labels, .. by hand.

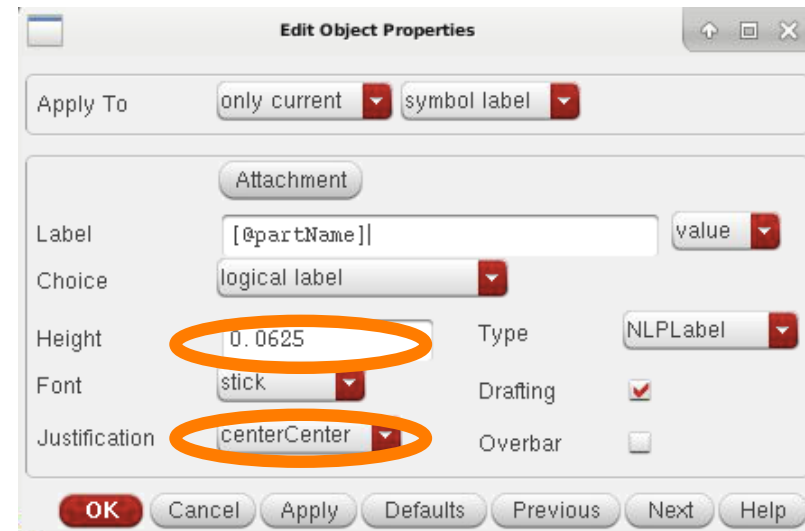


Editing a Symbol

- When you (later) add new pins to the schematic, you **also** have to add them to the symbol.
 - Make sure **name** and **type** are the same!
 - Best copy other pins and rename them



- You can move, stretch, ... as usual
- You can change the size or 'justification' of the labels





@instanceName and @partName

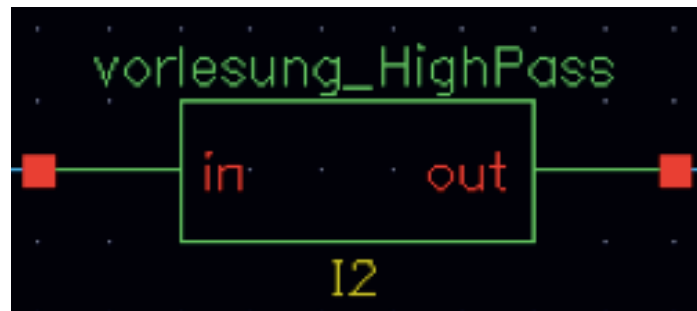
- Two special labels are created automatically:

`[@instanceName]`



do not
confuse with
`@cellName`

- `[@instanceName]` will display – surprise! – the name of the instance (of this symbol) that you place in another schematic, i.e. **I2** or, better, **lamp1** or so
- `[@partName]` displays the (library) name of the cell, i.e. **vorlesung_HighPass** or **NAND2**



- Place them somehow **nicely** (size / alignment / position)



The Selection Box

- When created automatically, a (red) *Selection Box* appears
- It marks the area which will be used to 'highlight' / 'select' the instance (in the next hierarchy level):



Selection Box
in **symbol**



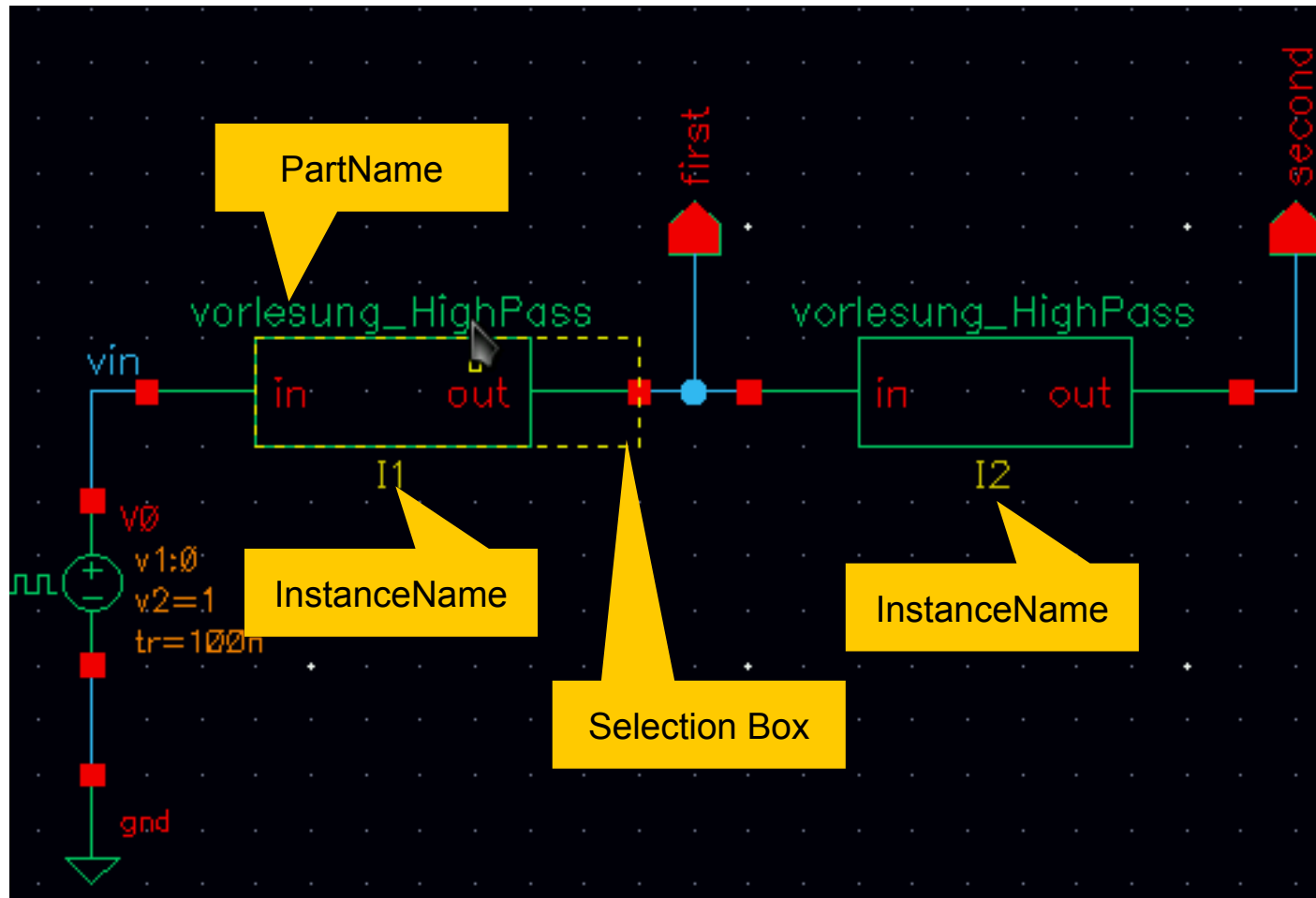
Highlighting in a
schematic which uses
an instance of the cell

- The *Selection Box* can be moved / resized
- If lost (or in manually created cells), it can be created by **Create** → **Selection Box**
- You cannot route over the Selection Box → keep it small
- If no Selection Box is defined, the maximal symbol size is used.



Using the symbol

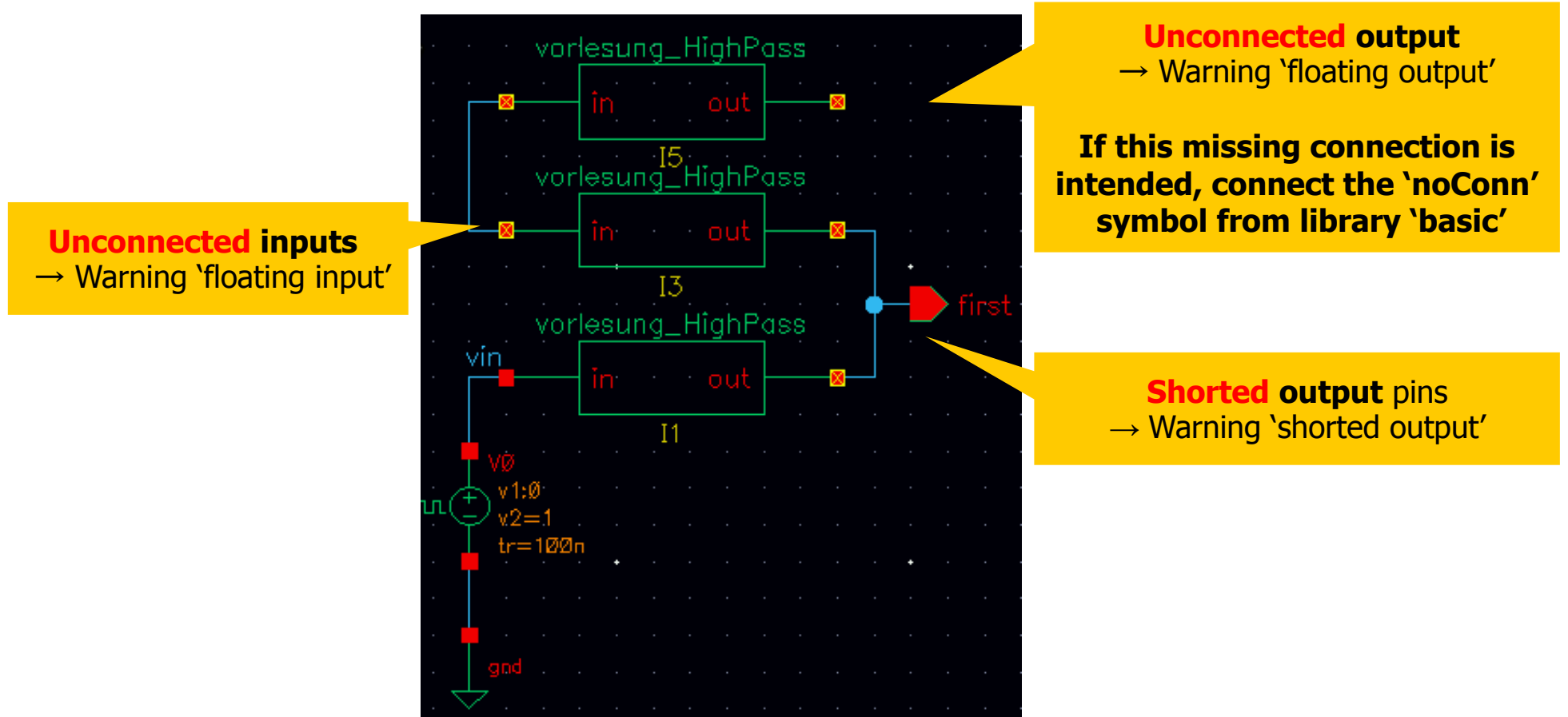
- In a schematic, you can add your symbol now in the same way as any other instance





Inputs / Outputs / InputsOutputs

- After 'Check & Save', warnings may pop up in the CIW:



- InputOutputs can be connected arbitrarily. Use with caution!
- All schematics should be 'clean', i.e. issue no warnings!



TRAVELING THE HIERARCHY



Traveling in the Hierarchy

- Assume you are in Schematic A which contains an Instance of PartType B
- If you want to modify (the symbol or schematic of) B, you normally have to open that cell from the library browser

- You can better *'dive into'* B by

- Selecting the instance
- Edit → Hierarchy → Descend Edit (Shift-X)
- Select the view
- Select if you want a new window / new tab / use existing tab



- You then end up in symbol / schematic of B
- When done, return back 'up' with Edit → Hierarchy → Return (Shift-B)
- You can also Descend for Read Only (Ctrl-X) or Edit in Place (x). This Edits B but shows A ! **Powerful but dangerous!**



GLOBAL NETS



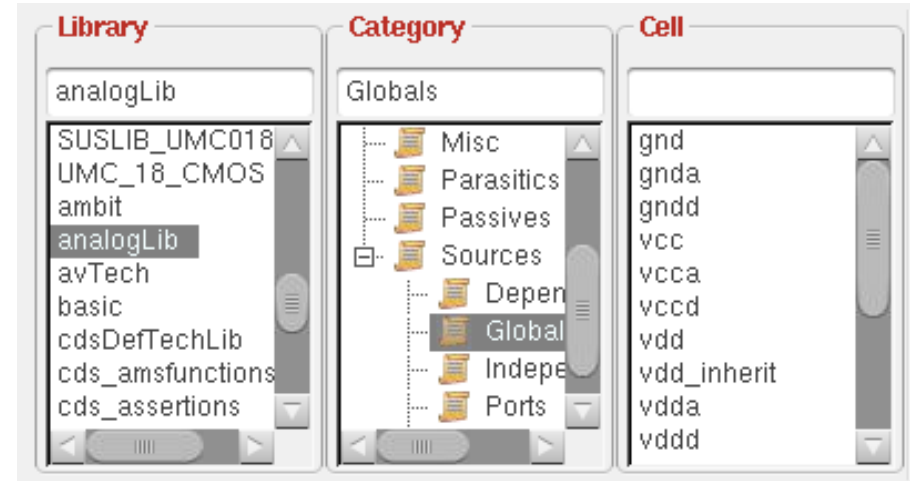
Global Nets

- A net is normally only known in the corresponding schematic
 - Connecting nets *between* schematics requires *pins*
- This can be tedious for signals which are used very often
 - analogue / digital power / ground
 - substrate potential
- You can use **global** nets, known **everywhere**
 - They are identified by an **exclamation** mark: xxx!
- Common global nets are
 - gnd! or sub! chip substrate
 - gndd! and vddd! digital ground /supply
 - gnda! and vdda! analogue ground / supply
- Handle them with care, because it is hard to track where they are used...



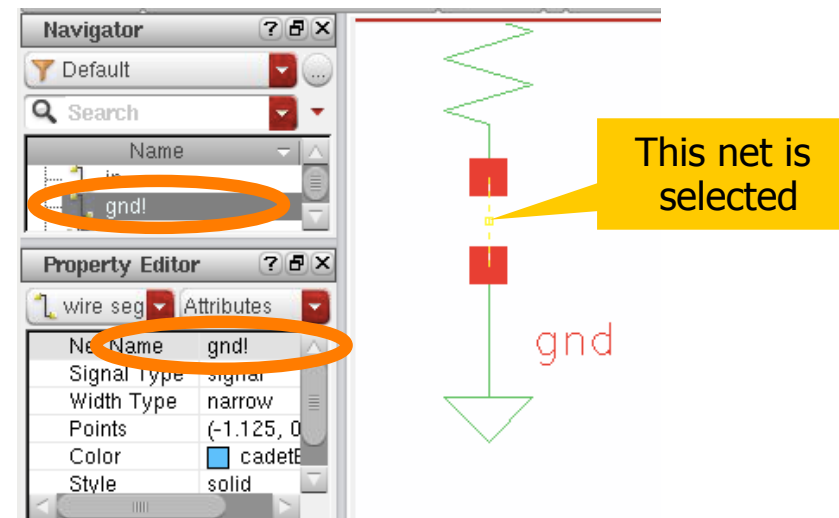
Global Nets

- There are several global 'symbols' in analogLib
 - Under Sources → Global



- They connect a net automatically to the corresponding global net

- Therefore:
Connecting to symbol 'gnd' is the same as labelling a net with 'gnd!'

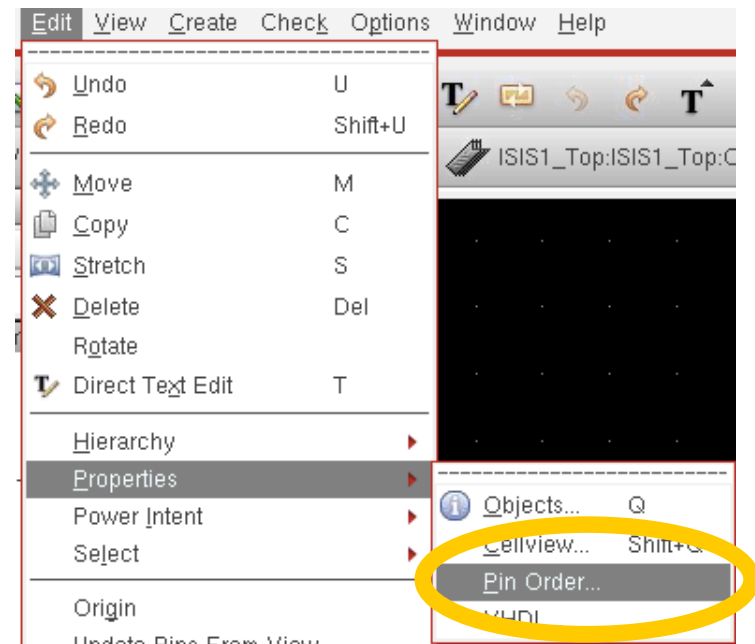
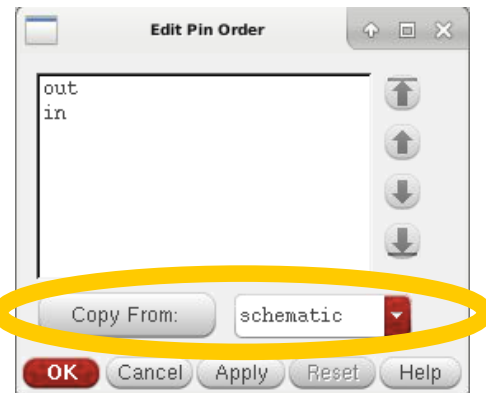




Pin Order

- If can happen that the internal order of pins gets messed up
 - You get a warning at Check & Save
 - This happens if you copy pins from other cells, delete pins,..

- To restore correct order, use **Edit → Properties → Pin Order**
- Best copy the Pin Order from another view:



- In rare cases, you have to regenerate (for instance) the symbol. (There is a step which allows you to just 'repair' the wrong stuff so that you nice drawing is not affected)



BUSSES AND ADVANCED NET NAMING



Advanced Net Names (Important!)

- A single 'wire' on the schematic can represent **several** nets, i.e. it can be a 'bus' or bundle of nets.

When a wire has multiple nets assigned:
Imagine the nets **stacked onto each other in the order they are listed**

- **Examples:**

- Simple wire `in`
- Multiple wires `a,b` separated by **comma**
- Bus `d<4:0>` 5 signals: `d<4>,...,d<0>`
- Bus `x<1:5>` different index order: `x<1>,...,x<5>`
- Repetition `<*3>a,<*2>b` this is equivalent to `a,a,a,b,b`
- Skip indices `d<7:3:2>` = `d<7>,d<5>,d<3>`
- Index list `d<1:0,3,<*2>5>` = `d<1>,d<0>,d<3>,d<5>,d<5>`
- Grouping `<*2>(a,b)` = `a,b,a,b`

- This works for **labels** and for **pins** (but better only use busses!)



More Complicated Examples

- You often need a 'binary' encoding of signals:
 - $\langle^*4\rangle(a,b) =$
 $\langle^*4\rangle(\langle^*1\rangle a, \langle^*1\rangle b) = a,b,a,b,a,b,a,b$
 - $\langle^*2\rangle(\langle^*2\rangle a, \langle^*2\rangle b) = a,a,b,b,a,a,b,b$
 - $\langle^*1\rangle(\langle^*4\rangle a, \langle^*4\rangle b) = a,a,a,a,b,b,b,b$



Advanced Net Names

- If you are not certain how a complicated net name expands:
Type the **expression** in the CIW (Command Interpreter Window) using

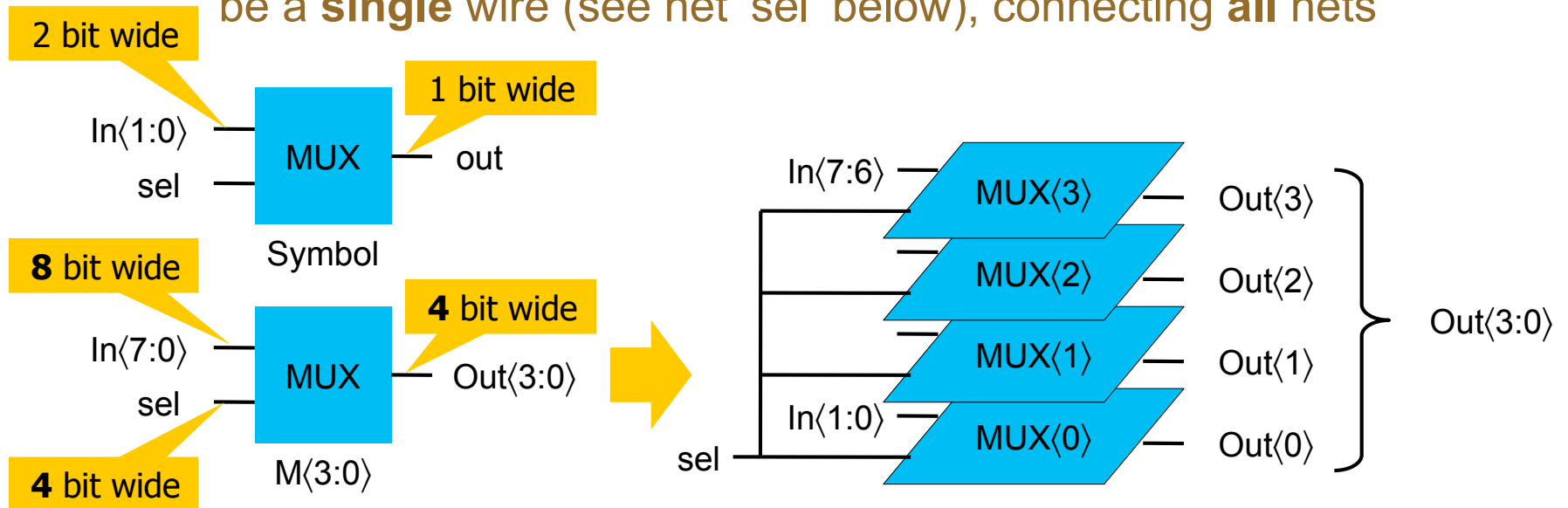
(dbProduceMemName "expression")

```
Virtuoso®  
File Tools Options Help  
(dbProduceMemName "<+2>a<1:0>")  
("a<1>" "a<0>" "a<1>" "a<0>")  
  
(dbProduceMemName "<+2>a<1:0>")  
|
```



Multiple Symbols (Very Important!)

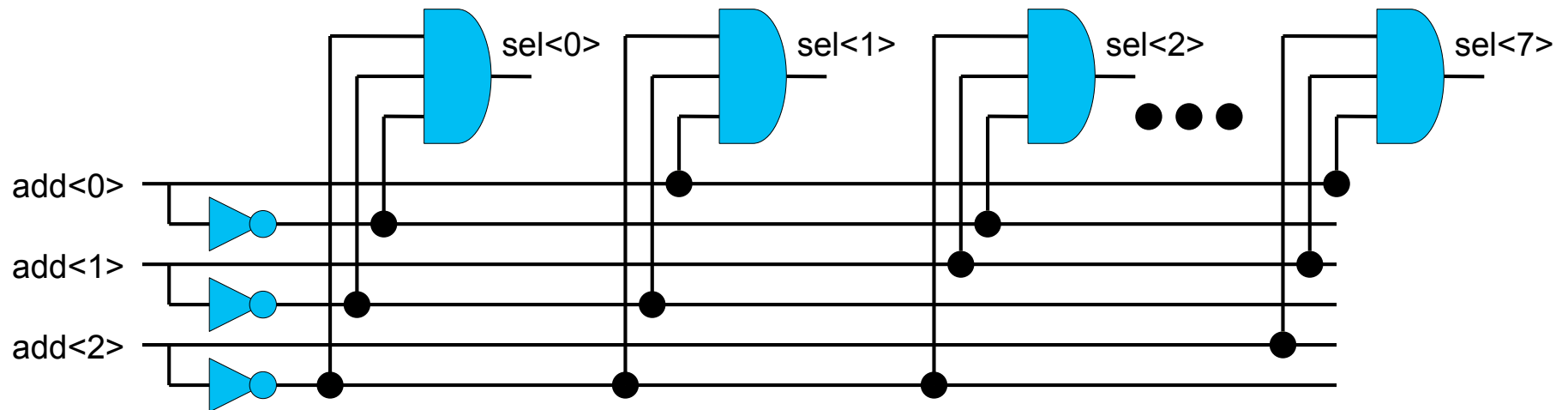
- **Instances** can be indexed as well:
 - An **instance** with name $M\langle 3:0 \rangle$ contains 4 elements $M\langle 3 \rangle \dots M\langle 0 \rangle$
 - They are (again) lying 'on top of each other' (in the order given)
- The instance **pins** are also stacked on top of each other
 - A single pin of N instances becomes a bus which is N nets wide
 - A pin with 2 nets ($in\langle 1:0 \rangle$) becomes $2N$ nets wide etc.
 - Connected nets must be have **exact** length OR be a **single** wire (see net 'sel' below), connecting **all** nets



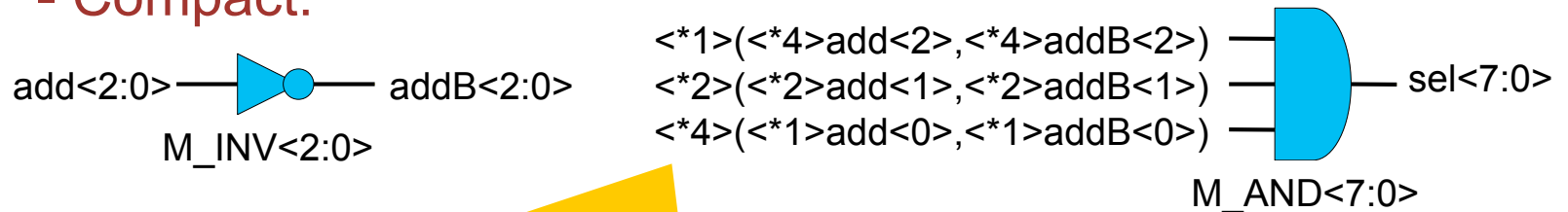


Multiple Symbols: 2nd example

- Here is a 3 Bit address decoder which activates one of 8 output signals $sel\langle 7:0 \rangle$ as a function of 3 address inputs $add\langle 2:0 \rangle$:



- Compact:**



This is: $add\langle 0 \rangle, addB\langle 0 \rangle, add\langle 0 \rangle, addB\langle 0 \rangle, add\langle 0 \rangle, addB\langle 0 \rangle, add\langle 0 \rangle, addB\langle 0 \rangle$



SYMBOLS WITH PARAMETERS

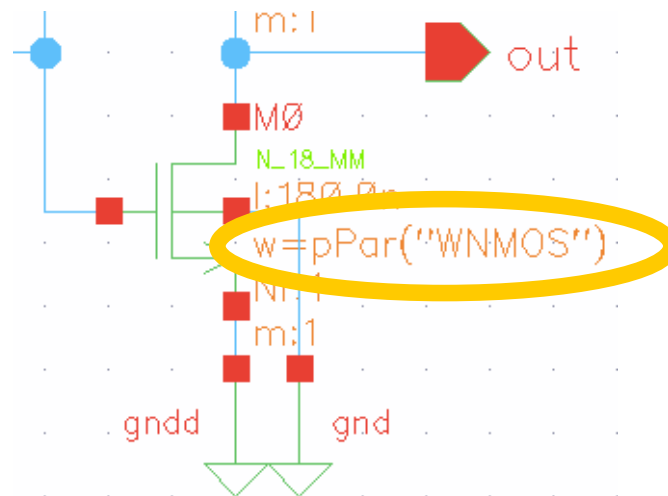


Parameterized Symbols (Step 1/3)

- It occurs that you need very similar schematics where only few parameters are changed (often transistor sizes)
 - Example: Inverter with different PMOS widths
 - (Unfortunately, parameters cannot be used everywhere...)
- Instead of creating multiple cells, you can create one cell with a PARAMETER:

1. In the schematic:

introduce the parameter with **pPar("pname")** (capital P!)



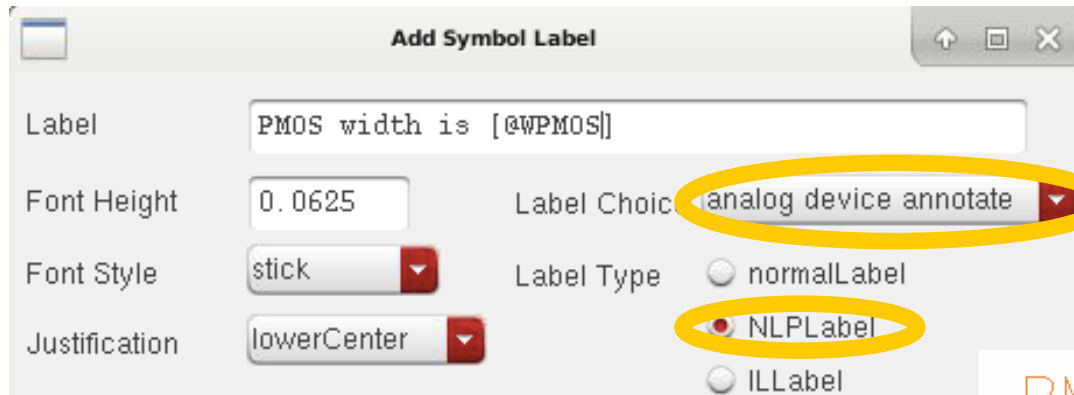


Parameterized Symbols (Step 2/3) – Optional!

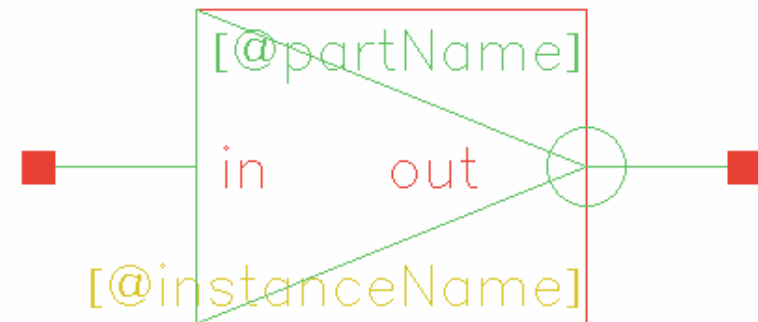
2. In the symbol: Add a label

- Label Choice: *analog device annotate*
- Label Type: *NLPLabel*

Add any text, referring to the parameter as `[@pname]`



PMOS width is [@WPMOS]

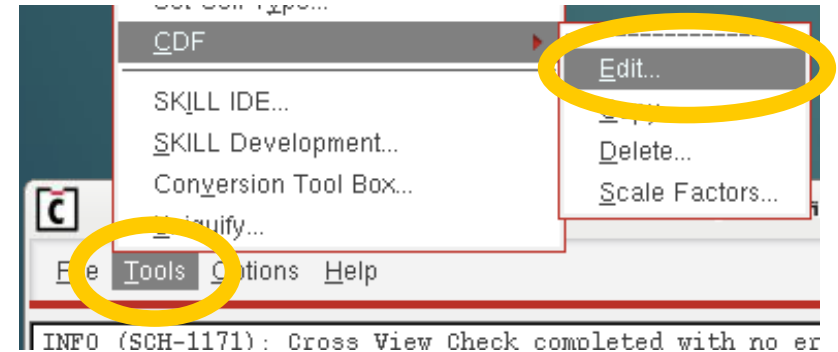




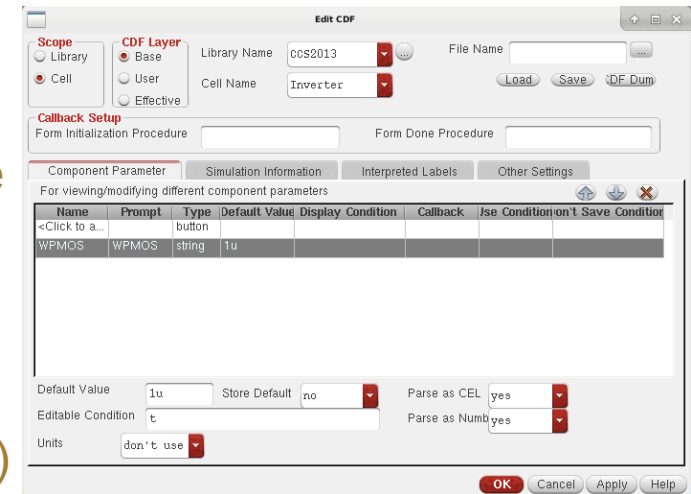
Parameterized Symbols (Step 3/3)

3. Cadence still needs to know about the new parameter:

- In **CIW**→**Tools**→**CDF**→**Edit**
- Choose Scope: Cell
- Choose CDFLayer: Base
- Select Cell



- Add your *pname* in the form
 - Type: String
 - Set prompt string & default value
 - Store Default: no (=default)
 - Parse as CEL: yes
 - Parse as Number: yes
 - Editable Condition: t (needed ?)
 - Units: don't use (=default)

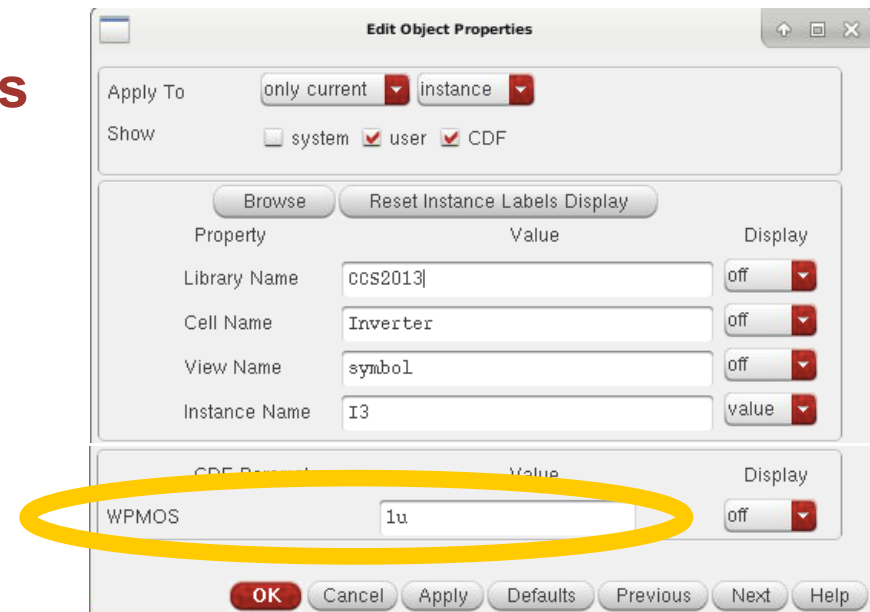
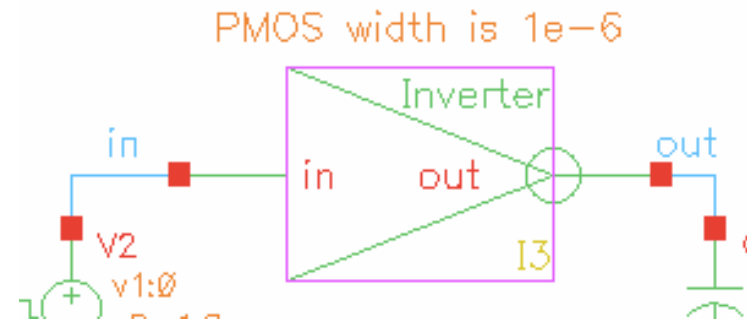




Parameterized Symbols: Instantiation

- The Symbol now shows your text + value
 - You may need to delete and re-instantiate existing symbols..

- You can now change the parameter in the instance **properties**





INHERITED NETS



Inherited Nets

- It is possible to over-write nets in schematics (mostly supplies) from a higher hierarchy level.
- This 'inherited nets' approach is not further described here...



FEATURES OF THE SCHEMATIC EDITOR



Some Features

- The additional display of net names at the pins of transistors is often confusing.
 - You can turn this off under **View** → **Hide Terminal Labels**
- If you want to see all places where a net in one schematic connects:
 - Enable Highlighting under **View** → **Net Highlighting**
- If you want to follow a signal through the hierarchy:
 - Highlight the net under **Create** → **Probe** → **Add Net**