



advico®
Design Conversion Tools Manual

050523

Table of Content

What's that?.....	3
ShrinkCell.....	4
ShrinkLib.....	5
ShrinkMOSpCell.....	6
UpdateMosCdf.....	7
UpdateResCdf.....	8

What's that?

advlCo's Design Conversion Tools have been created to save your (and our) time when updating parameters for a designed circuit (e. g. when you designed with a technology under development) or transferring a design from one technology to another. The tools work with **advlCo's** design kits for IHPs technologies and are developed and maintained on demand – ask us if you need a conversion which is not listed here.

Some functions come with a GUI (Graphical User Interface), others are used from the Cadence CIW command prompt. Since all conversions deeply modify schematics and / or layouts, we strongly advise you to make a backup of your entire design before conversion.

ShrinkCell

Function Description

The ShrinkCell function will shrink a single layout cell by a specified factor. Other layout cells which are inserted will not be modified.

Execution:

Open any window of the desired library, make it active and call ShrinkCell(cell view scale plain) to scale a single cell. Cell and view are names which must be given in double quotes. Scale must be given as a float. To convert from IHPs SGC25C process to IHPs SG25H1 process, scale is 0.7. The option plain (t/nil) determines if instances in the layout cell are also scaled (nil) or not (t). You must specify t if the cell contains mosaics. Pcells in the layout will not be transformed because special handling of the individual pCell parameters is required. Pcell conversion can be done manually or with dedicated functions. If the design is not on the production grid after shrink, you may want to use our DRC tools, especially the onGrid GUI, to correct them automatically within a glimpse.

ShrinkLib

Function Description

The ShrinkCell function will shrink a layout library, e. g. a complex hierarchical design, by a specified factor. Only cells within the selected active design library are modified.

Execution:

Open any window of the desired library, make it active and call ShrinkLib(view scale) to scale all views of the opened library. View name which must be given in double quotes. Scale must be given as a float. To convert from IHPs SGC25C process to IHPs SG25H1 process, scale is 0.7. The function will search for any views with the specified name and execute a plain shrink on each cell. The result is a completely shrunked hierarchical layout drawing – only pCells must still be treated manually or by other functions.

ShrinkMOSpCell

Function Description

The ShrinkMOSpCell function performs the operation of scaling the parameters of nmos and pmos pCells, thereby changing schematic, layout or both views of your design. This gives you the possibility to convert a complete complex CMOS design to a different technology and have simulation and layout ready within the shortest possible time.

Execution:

Open any window of the desired library, make it active and call ShrinkMOSpCell(scale) to open the GUI. The GUI will ask if schematics, layouts or both should be translated. Existing instances from illegal device libraries will be changed to instances of the valid technology library (the valid technology library is specified by attaching the design library to the technology library). No callback functions are called by this procedure. You should check if device parameters are valid (e. g. Not exceeding parameter boundaries) after conversion.

UpdateMosCdf

Function Description

The UpdateMosCdf function performs the operation of checking the parameters of nmos and pmos pCells, thereby changing schematic, layout or both views of your design. This gives you two benefits:

If you designed a chip in an early development state of a technology or in a different technology option, you may change the properties of all devices to reflect the behaviour of producing this design with new process parameters. In other words, you may simulate how a design with a given layout would perform in a different technology.

If you want to transfer a design to a different process option or if process parameters have changed, it is a lot of work to rescale all devices to keep the desired electrical behaviour. The UpdateMosCdf function will also allow to keep the main schematic parameters and change dependent schematic and layout parameters to simplify the update process.

Execution:

Open any window of the desired library, make it active and call UpdateMosCdf() to open the GUI. The GUI will ask if schematics, layouts or both should be translated. Existing instances from illegal device libraries will be changed to instances of the valid technology library (the valid technology library is specified by attaching the design library to the technology library). No callback functions are called by this procedure. You should check if device parameters are valid.

UpdateResCdf

Function Description

The UpdateResCdf function performs the operation of checking the parameters of resistor pCells, thereby changing schematic, layout or both views of your design.

This gives you two benefits:

If you designed a chip in an early development state of a technology or in a different technology option, you may change the properties of all devices to reflect the behaviour of producing this design with new process parameters. In other words, you may simulate how a design with a given layout would perform in a different technology.

If you want to transfer a design to a different process option or if process parameters have changed, it is a lot of work to rescale all devices to keep the desired electrical behaviour. The UpdateResCdf function will also allow to keep the main schematic parameters and change dependent schematic and layout parameters to simplify the update process.

Execution:

Open any window of the desired library, make it active and call UpdateResCdf() to open the GUI. The GUI will ask if schematics, layouts or both should be translated. Existing instances from illegal device libraries will be changed to instances of the valid technology library (the valid technology library is specified by attaching the design library to the technology library). No callback functions are called by this procedure. You should check if device parameters are valid.

Note: The next release of the CDF conversion functions will execute the original callback functions, allowing enhanced consistency in case of conversion problems.

Disclaimer

- 1) advlCo warrants that the programs conform with the above specifications. However it is expressly stated that the current source and binary representation of the programs may not exclude the occurrence of defects or malfunctions under certain circumstances.
- 2) advlCo will not warrant the licensed programs if the customer modifies the licensed programs or damages the media upon which the licensed programs are supplied.
- 3) If any defects cannot be remedied at reasonable expense, advlCo is entitled to terminate the license for the licensed program concerned and refund the license fee.
- 4) The right to request remedy of a defect shall lapse 6 months after shipment of the Licensed programss to Customer.
- 5) advlCo shall be liable for damage to the customer caused by defects and malfunctions in the Licensed Program only in case of gross negligence and wilful intent. advlCo shall be liable for direct damages caused by slight negligence up to the amount of the license fee.