VCR Tutorial

This application note outlines the use of the Virturso Custom Router tool (also known as the Cadence Chip Assembly Router) for use in routing of datapath blocks, datapath assembly, and top-level chip routing.

Design Library Setup

1. Create a directory, *test_router* in your eecs427 directory.
2. Copy the file /afs/umich.edu/class/eecs427/tsmc25/ccar.rul into your *test_router* directory
3. Copy the file /afs/umich.edu/class/eecs427/tsmc25/dofile.do into your *test_router* directory
4. Copy the directory /afs/umich.edu/class/eecs427/tsmc25/test_router.tar into your test_router directory.
5. Untar the test_router.tar file using "tar –xvf test_router.tar " from your command prompt while withing the test_router directory
6. Prepare the test_router directory to launch icfb from (like you did for cad1 and cad2).
7. Run icfb.

Editing your cds.lib

Once icfb has launched, you need to link the test_router directory to your techfile and enter the path of the directory into your cds.lib. This can be done readily by:

1. In the CIW, click Tools -> Library Manager
2. In the Library Manager, click Edit->Library Path
3. This opens up the Library Path form. Click on an empty entry in the form, enter "test_router" for the library field and in the path field, enter the full path for the test_router design library – "/afs/umich.edu/class/eecs427/w07/<username>/test_router"
4. In the Library Path form, click Edit->Exclusive Lock, and confirm.
5. Finally, click File-> Save.

This is a way to enter the path of your design library into cds.lib so that it can be accessed by the Library Manager.

Virtuoso XL

Now you are ready to open the test_router design library. You will be prompted for the techfile to be attached to this library. Attach the same techfile you have been using so far. Open the schematic called test_router. Inspect this schematic. You should see a 13-bit input bus providing inputs to nand and nor gates. These gates have been provided for you within the test_router directory for convenience. Now, we will map this schematic to the layout as follows.

1. Click Tools-> Design Synthesis->Layout XL
2. You may be asked if you would like to create a new layout view or replace an existing view. Select "create new"
3. Choose "layout_placed" in the viewname field of this form.
4. A layout window will pop up. This window is a Virtuoso XL window, very much like Virtuoso, but with the added benefit that if you name the layout cells to be the same as the schematic instances, the layout view will indicate to you how cell pins in the layout connect to

each other. This will be indicated by a “rats netlist”, which are little lines that appear if you try to move one of the layout cells.

5. Click Design->Gen From Source and a layout generation options form will pop up. This form helps you create layout instances with the same names as those in your schematic and create pins which you will use to guide your routes from this block to a later block you will connect to.
6. Make sure that the radio buttons corresponding to Instances and I/O pins are selected. Unselect the Boundary button. You will create your own in the prBoundaryLayer
7. In the I/O pins section **for this tutorial** , select the layer/master button to metal3. Then select all the pins in the window and click “Apply”. This will create all pins in metal3.. You will have to change these to metal1 pins and duplicate them to add them to all the horizontal power lines.
8. The cells will now be found scattered all across the screen. Pins wile be found just a small distance to the bottom and right of the origin. You are responsible for the appropriate placement of the cells and the pins.
9. Close the *layout_placed* view.
10. For the purposes of expediency, we have prepared a sample layout scenario for you in the *layout_vssathe* view. When you are done placing the cells, they may end up looking something like this. In this tutorial, the pins are on the side of the block. Keep in mind that in your datapath design, pins will be found on the top and bottom of the block. Also note that the pins are placed on the horizontal metal3 track. This is vital for routability given the rules that will be adopted for the router.
11. Save the *layout_vssathe* view as *layout_placed,* thereby overwriting the design that contained the randomly distributed layout cells. In future designs, this will contain the placed design that you will create. Once you have a placed design, (which in this tutorial is provided by us), make a copy of this design into another view *layout_routed.*
12. Again in the schematic window click Tools -> Design Synthesis -> Layout XL, and this time select “open existing”. This will map the schematic to an already existing cell view.
13. Select layout_routed cell view. This view will contain the routed layout after you are through with the router. Notice that when you select a cell in the layout view, the corresponding cell lights up in the schematic view.
14. Now you are ready to route. Click on Routing->Export to router. This opens up an export form.
15. Make sure the Export Layout Cellview is correct. You don’t need to enter anything in the Export Netlist Form since Virtuoso XL already has the mapping information of which cell terminals are to be connected together.
16. Click on “use rules file” and select ccar.rul, the file you copied from the tsmc25 directory.
17. Finally, at the bottom of the form select the cadence chip assembly router radio button in the Routers section.
18. The router window should automatically show up.

VCR- Virtuoso Custom Router

1. Now open up the dofile.do in your favourite editor. This is a barebones dofile which will let you complete simple routing tasks. We urge you to look at the commands and try to understand what they do. A useful reference on the commands used for vcr (also known as ccar) is cdsdoc (icc11* section) on the virtuoso router.

2. Simply enter the commands in the dofile into the command field at the bottom of the router window. You may choose to cut and paste the commands into the command field or enter them one by one to see what happens.
3. After the “fix net vdd! gnd!” command, which incidentally tells the router to not route the vdd! and gnd! nets. Enter “route 20; clean 10”. This command runs 20 passes of routing (dirty, can have DRC violations) followed by 10 passes of clean routing, which will try to fix up all those violations. You will see the wires being routed in your netlist automatically. In the bottom of the router window, a pane displaying the progress of your routing job is updated. It reports the number of conflicts you still have in your route. Square boxes are drc/lvs violations and diamonds are certain lvs violations. To get a list of all conflicts, use the report constraints command at the end of the dofile (It has been commented).
4. You should have no spacing or cross (short-circuit) violations.
5. You may have length conflicts. These conflicts arise due to the “circuit class length” rule which sets requirements on the length of the route. As such, this conflict will not lead to DRC or LVS errors.
6. If you have some violations they may be due to notches. These are same_net violations that can exist. Use the “remove notch” command to remove these
7. If you still have have some spacing violations, they may be because of congestion in metal3. Use the “delete all nets” command and repeat the routing and clean command.
8. If there are still violations, relax the metal 3 track rule. You will notice that in your design, signals move vertically in the design so vertical tracks should be kept as strictly as possible. You have some flexibility with the horizontal metal3 tracks. So you can relax the constraint using the grid wire command (last command in dofile.do. It has been commented out).
9. This command sets the routing tracks in metal 3 to be of very fine granularity, allowing the metal3 wires to “slightly adjust” to make space for the spacing violations.
10. Delete all nets in the router and run the route and clean commands again. DRC should pass. If problems persist, contact your GSI.
11. Now save the design. Click File->Write->Session and let the file be the default .ses file and quit the router.
12. In icfb, you will find that the wires have been imported. If not, just click on Router->Import from router and use the .ses file you saved to.
13. Now extract the layout, clicking on the “Join nets by name” button.
14. Run LVS. You MAY still have to fix a few DRC errors but as we perfect this flow, this wont be a problem either.