Layout Generation

Carsten Wulff, carsten@wulff.no

Keywords: Layout Generation, CICPY, Placement, Routing, Automation

Layout

The open source tools don't have any automatic analog layout. To my knowledge, there is no general purpose analog automagic layout anywhere in the world. It's an unsolved problem. Many have tried (including myself), but none have succeeded with a generic analog layout engine.

There are a few things, though, that could help you on the way.

Setup

I assume that you have the latest and greatest `aicex\ip` setup.

See SKY130NM Tutorial if aicex is unfamiliar.

Let's assume we use `jnw_gr05_sky130a` to test out our layout

```
cd aicex/ip/
cd jnw_gr05_sky130a
git checkout a1e3dfc324194729e042f5e653777b052759863b
cd work
```

CICPY

The first thing we need to do is to place all transistors. I do have a script to help. Install cicpy.

```
cd aicex/ip/cicpy
git checkout master
git pull
python3 -m pip install -e .
cd ..
cd cicspi
git checkout main
git pull
python3 -m pip install -e .
```

Placement

To generate an initial placement we can do the command below. If a layout exists it will be overridden

```
cd jnw_gr05_sky130a/work
cicpy sch2mag JNW_GR05_SKY130A OTA_Manuel
```

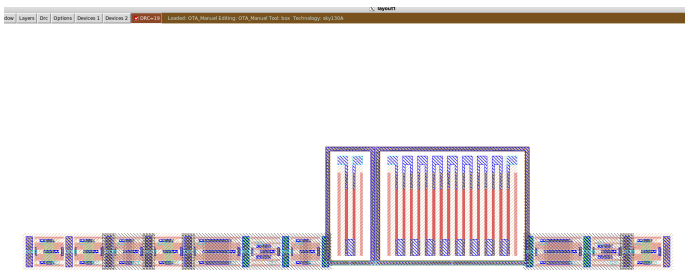

Figure 1: Initial cicpy sch2mag placement of OTA_Manuel in Magic - all transistors in one row, 19 DRC errors

The layout engine has no idea what components belong together, for example, the current mirror below should have been placed together

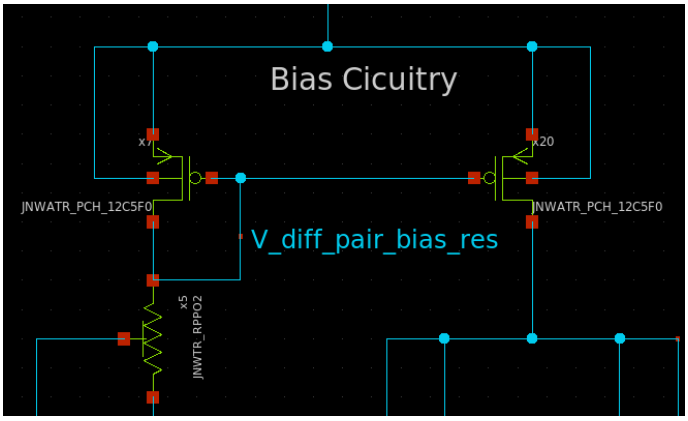

Figure 2: Xschem schematic of the OTA bias circuitry, where the current mirror pair xa07/xa20 should have been placed together

We can instruct the layout engine by adding a "group" name to the instance name. The instance name always starts with `x<something><number>` where the something can be nothing, or a group name (a,b, not a number).

The rules for placement are:

1. Sort all instances by groups
2. Sort all groups by instance name
3. Place the first instance.
4. For all instances: If the next instance has the same group, then add on top. Otherwise increment the x location.

As such, if I rename my instances, as shown below,

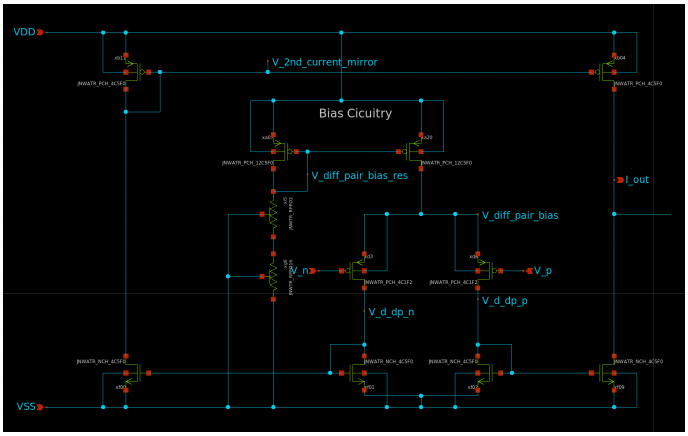

Figure 3: The same OTA schematic after renaming instances with group prefixes (xa, xb, xd, xf, xg) to guide the placer

Then the layout becomes a bit better

```
cicpy sch2mag JNW_GR05_SKY130A OTA_Manuel --gbreak 3 --xspace 34000 --yspace 30000
```

The gbreak command inserts a “group break” after the fourth group, such that a new Y coordinate is selected.

The X and Y space is for the distance between groups. The unit is “Ångstrøm”, so 1 um is 10 000 Å.

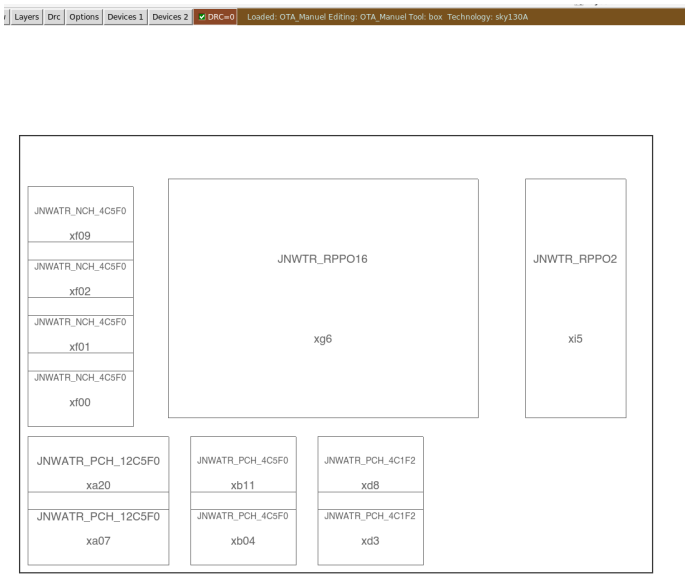

Figure 4: Placement after grouping, shown as instance boxes: devices of the same group stack vertically, and –gbreak 3 starts a new row

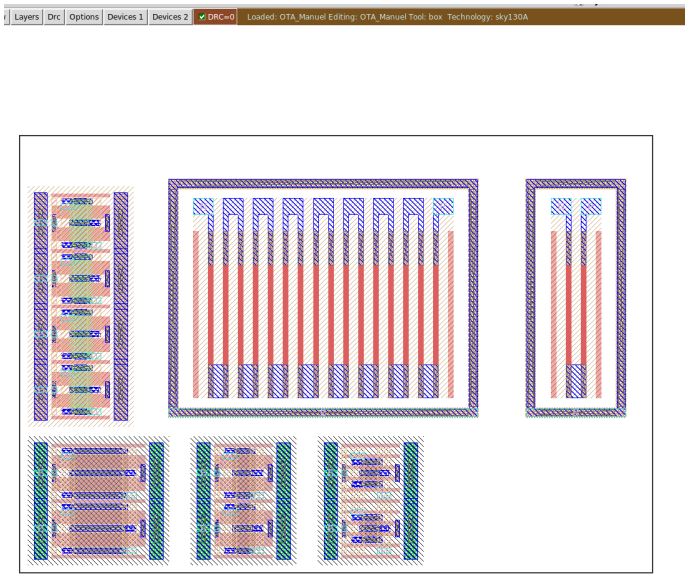

Figure 5: The same grouped placement with all layers drawn, DRC clean

SUMMARY

The one-page version of this chapter:

- Layout is the final translation: from schematic to the polygons the fab will etch
- Matching is geometry: symmetry, proximity, dummies, and common centroid where it counts
- The parasitics are part of the circuit - extract and re-simulate before believing any layout
- DRC and LVS are not suggestions: clean both, every time

WOULD YOU LIKE TO KNOW MORE?

Hastings, *The Art of Analog Layout* - the book on matching, and worth owning

The Magic and netgen documentation, which is the practical companion for the flow used here

Carsten Wulff received the M.Sc. and Ph.D. degrees in electrical engineering from the Department of Electronics and Telecommunication, Norwegian University of Science and Technology (NTNU), in 2002 and 2008, respectively. During his Ph.D. work at NTNU, he worked on open-loop sigma-delta modulators and analog-to-digital converters in nanoscale CMOS technologies. In 2006-2007, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. Since 2008 he’s been with Nordic Semiconductor in various roles, from analog designer, to Wireless Group Manager, to currently Principle IC Scientist. From 2014-2017 he did a part time Post.Doc focusing on compiled, ultra low power, SAR ADCs in nanoscale technologies. He’s also an Adjunct Associate Professor at NTNU. His present research interests includes analog and mixed-signal CMOS design, design of high-efficiency analog-to-digital converters and low-power wireless transceivers. He is the developer of Custom IC Compiler, a general purpose integrated circuit compiler, and makes the occational video on analog integrated circuits at https://www.youtube.com/@analogicus. For full CV see https://analogicus.com/markdown-cv/.