Resolving IP cell-name conflicts peacefully

By Dennis Joseph | No Comments | Posted: October 26, 2020
Topics/Categories: IP - Assembly & Integration (https://www.techdesignforums.com/practice/topics/ip-topics/assembly-integration/), Design Management (https://www.techdesignforums.com/practice/topics/ip-topics/design-management/), EDA - Verification (https://www.techdesignforums.com/practice/topics/eda-topics/design-verification/) |
Tags: cell name conflict (https://www.techdesignforums.com/practice/tag/cell-name-conflict/), naming conflict (https://www.techdesignforums.com/practice/tag/naming-conflict/) | Organizations: Siemens EDA (https://www.techdesignforums.com/practice/organization/mentor/)

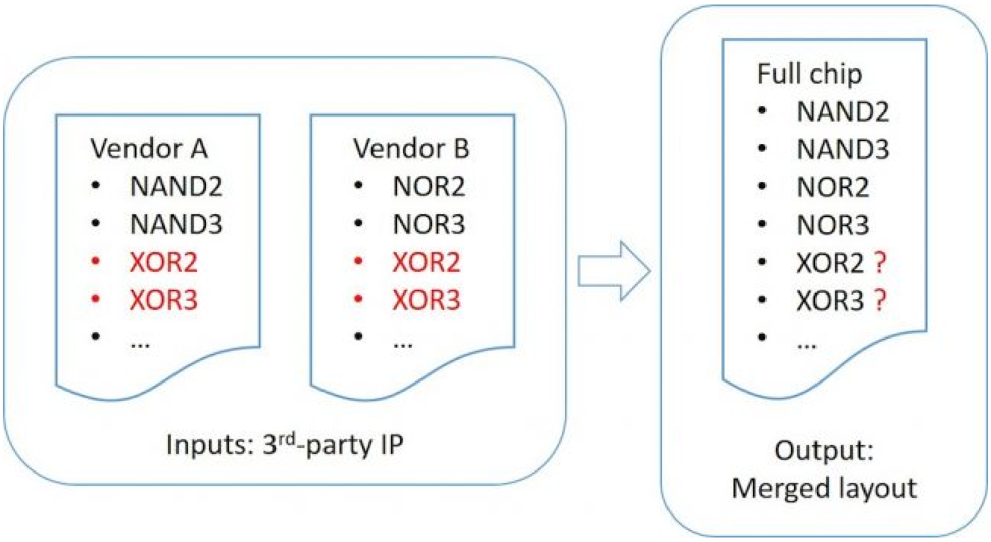
One roadblock to the integration of IP from multiple vendors into an SoC is the likelihood of finding duplicate cell names in the merged design. Carefully considered renaming strategies can fix the problem without causing design database bloat.

System-on-chip (SoC) designs are popular because they enable manufacturers to produce products that are smaller, cost less money to make, and require less power to operate. Chip design teams like SoCs because they can reduce their time to market by incorporating reusable intellectual property (IP) blocks into multiple SoC designs. Design teams take advantage of this opportunity two ways: they can implement some common functionality once for reuse in multiple designs, and/or they can purchase fully verified IP from 3rd-party suppliers for any number of functions. This off-the-shelf, reusable availability sets the SoC design teams free to focus on creating the proprietary IP that differentiates their designs, reducing the number of tasks and time needed to design an SoC and enabling teams to meet ambitious delivery schedules.

Of course, there are challenges to integrating 3rd-party content into a design. While each IP supplier verifies their IP at the block level, SoC design teams still have to successfully merge and verify all the different IPs at the full-chip level. One roadblock to this integration is the presence of duplicate cell names across all the IP blocks. Cell-name conflicts among IP blocks can result in multiple undesirable outcomes, including unexpected layout results at the full-chip level, larger file sizes that require more time and resources to manage, and the inability to run layout verification tools efficiently in hierarchical mode. Finding a way to effectively manage and resolve cell-name conflicts is an essential part of an accurate and efficient design flow.

Cell-name conflicts

Having just mentioned some of the adverse effects of cell-name conflicts, let’s take a closer look at how and why these conflicts occur. Assume a design team acquires two external IPs from different suppliers. Both IPs contain cells named XOR2 and XOR3. If designers don’t take these duplicate cell names into account when assembling the full-chip layout, the layout merge tools may not create the expected layout, and it may be nearly impossible to determine which version of the conflicting cells was written to the full-chip layout (Figure 1).

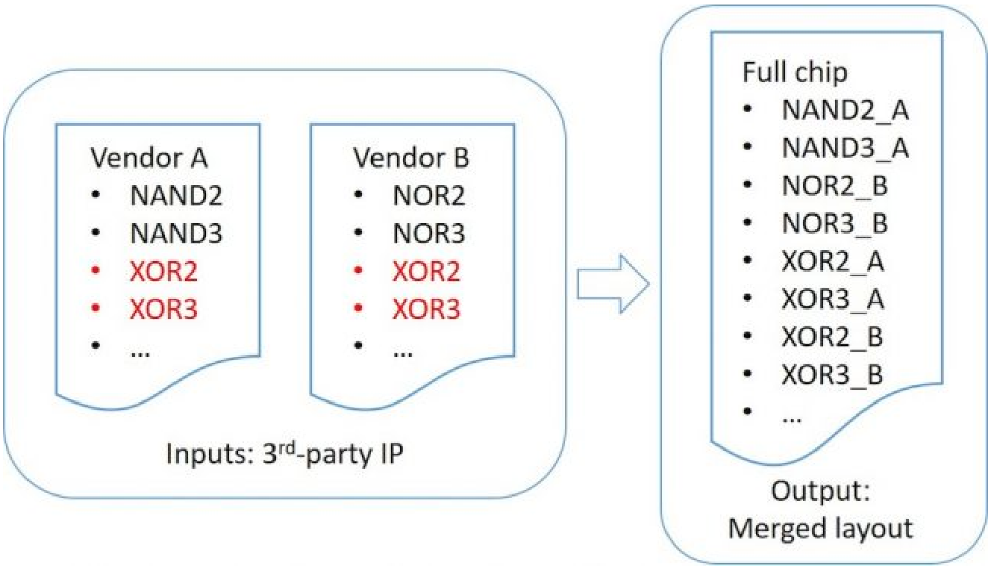


(https://www.techdesignforums.com/practice/files/2020/10/Fig1_Name-conflicts.jpg)

FIGURE 1 When IPs containing duplicate cell names are merged into a full-chip layout, it may not be clear which version of each cell was included

When physical and circuit verification tools use this merged data, they can end up generating meaningless error results. Unfortunately, this doesn’t just waste the time required for those runs. Because of the difficulty of determining the cause of the errors, designers can struggle to recognize that the results are meaningless. The additional time and resources needed to resolve the issues are wasted, and can extend tape-out schedules or reduce time available for other important tasks.

Some design teams avoid this scenario by uniquely renaming all cells in all input layouts prior to merging (Figure 2). While this approach does eliminate the possibility of cell-name conflicts, it also needlessly increases the merged layout file size (especially when the cells contain duplicate information). Renaming all cells also increases the runtime of many verification tasks, as layout tools can’t take advantage of the runtime efficiency a strong design hierarchy provides.



(https://www.techdesignforums.com/practice/files/2020/10/Fig2_Rename-all-cells.jpg)

FIGURE 2 Renaming all cells removes naming conflicts but increases file size and verification tool runtimes

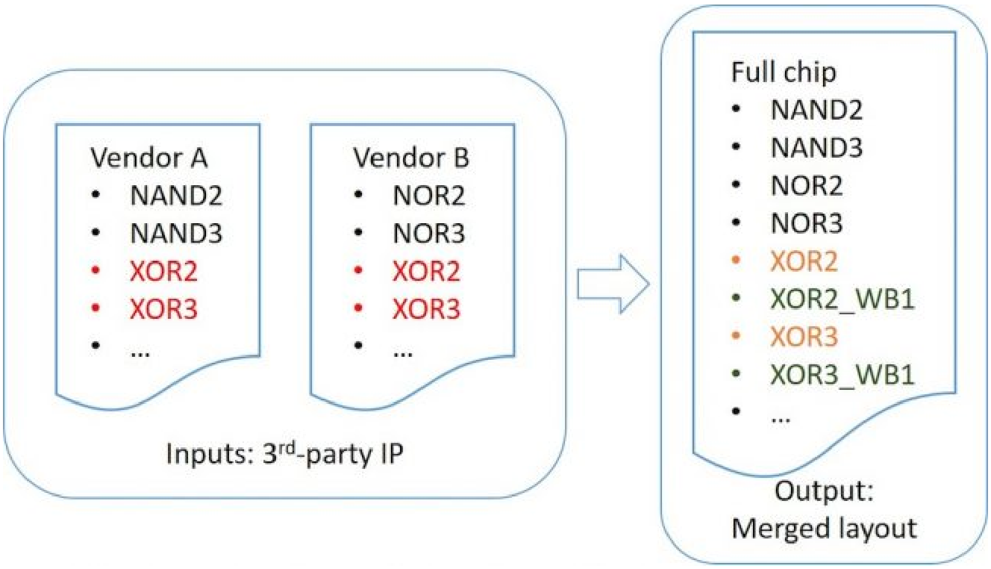
Automated cell-name conflict management

Fortunately, electronic design automation (EDA) companies have stepped up with automated solutions to provide a fast and efficient process for design teams to quickly and accurately resolve cell name conflicts. We'll look at how Mentor, a Siemens business implemented cell-name conflict resolution by introducing innovative functionality in its Calibre DESIGNrev chip finishing tool.

The Calibre DESIGNrev FileMerge functionality provides several levels of cell conflict management that enables design teams to create customized flows that deliver the level of detail they require. It reports all cell-name conflicts, which allows design teams to review discrepancies, follow up with their IP suppliers as needed, and rename cell-name conflicts as needed prior to merging for the best results.

Renaming conflicting cells during merge

Designers can easily and quickly rename all conflicting cells by running the FileMerge functionality in its rename mode. With the rename option, the output retains the original name for the first occurrence of a cell name conflict, renaming subsequent versions. Designers may choose a custom prefix or suffix to use when renaming cells to get a more meaningful name in the merged layout (Figure 3).



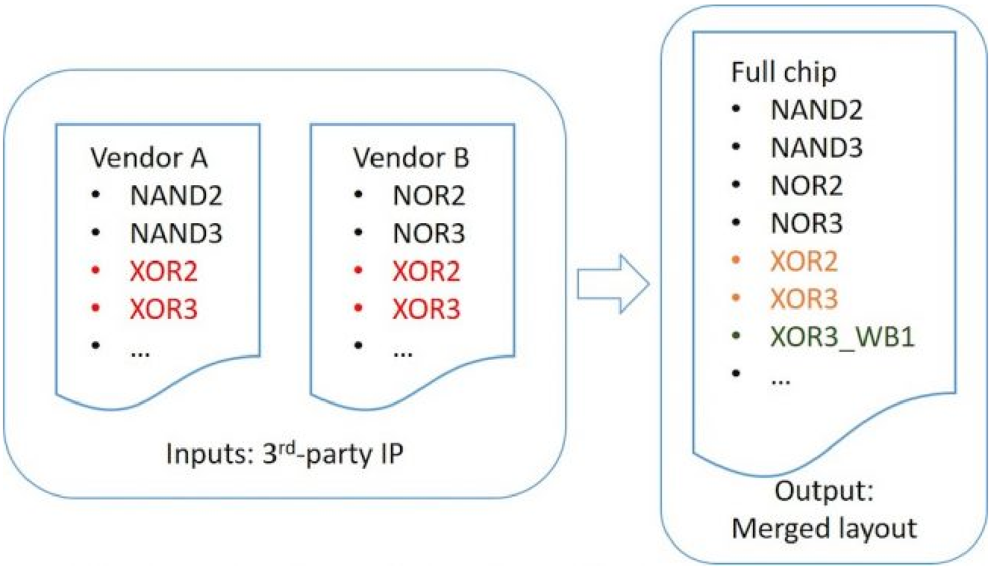
(https://www.techdesignforums.com/practice/files/2020/10/Fig3_Rename-only-conflicts.jpg)

FIGURE 3 The Calibre DESIGNrev FileMerge functionality lets designers quickly rename any duplicate cells

Only renaming the conflicting cells reduces the number of renamed cells in the layout, but can still result in a larger layout file than necessary, since some of those renamed cells have the same content as the cells that retain their original names. In addition, the source of a particular cell may still not be clear to designers viewing the merged layout results.

Comparing cell contents when resolving cell-name conflicts

One solution to the problem of duplicate content is to rename only those conflicting cells that actually contain different content. Adding a "smart" renaming option when invoking the Calibre DESIGNrev FileMerge functionality directs the tool to compare the contents of each cell in conflict, and then only rename those cells with actual content differences. In the example shown in Figure 4, the contents of XOR2 are the same in both IPs, so the cell name remains unchanged in the merged layout. However, because the contents of XOR3 are different in each IP, the second XOR3 cell is renamed.



(https://www.techdesignforums.com/practice/files/2020/10/Fig4_Smart-rename.jpg)

FIGURE 4 A smart renaming option ensures that only conflicting cells with actual content differences are renamed

To provide designers with very precise control over what is reported as a difference, the smart renaming option also has options that enable designers to ignore texts, ignore properties, and convert paths to polygons. With this level of control, designers can develop customized flows that ensure unique cell names are only applied when truly needed. This capability also ensures faster Calibre runtimes downstream, because the hierarchical Calibre engine runs more efficiently when it has fewer cells to process in a design.

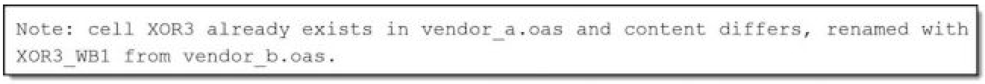
Cell-name conflict reporting

Despite all these options, designers may still not recognize the origin of renamed cells when they review the Calibre DESIGNrev FileMerge output. Even when designers specify a custom prefix or suffix, one version of the cells in conflict keeps its original name. Without knowing which cells from which input layouts were renamed, and what those new names are in the output, design teams may not have enough insight to evaluate the output and know that it contains what they expect.

To eliminate confusion, a report of all the cell conflicts provides this detail, and enables design teams to follow up with their IP suppliers in the event of any discrepancies. The Calibre DESIGNrev tool provides various reporting optionsto ensure that the merged output accurately reflects the design team’s intentions.

Reporting renamed cells during merge

By adding a conflict reporting option to the FileMerge invocation, designers can get a list of all the cell-name conflicts and their new assigned names. Using this reporting option with the smart renaming option provides a list of the cell-name conflicts in which the cell contents differ (Figure 5).



```
Note: cell XOR3 already exists in vendor_a.oas and content differs, renamed with
XOR3_WB1 from vendor_b.oas.
```

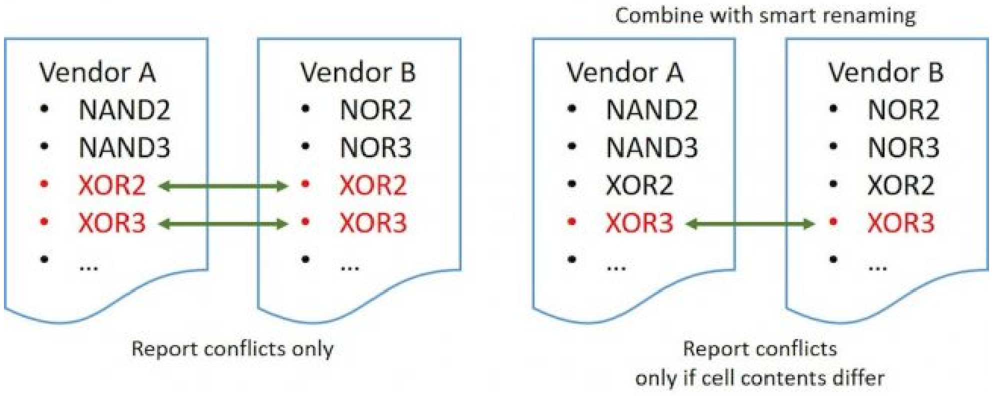
(https://www.techdesignforums.com/practice/files/2020/10/Fig5_rename-report-msg.jpg)

FIGURE 5 Using the smart renaming and conflict reporting options together identifies the cell-name conflicts and the new cell names assigned during the merge

This report enables designers to confirm that the output contains everything that is expected, and to follow up with their IP suppliers regarding any discrepancies. They can also update the hierarchical cell list with this information to ensure a fast and accurate layout vs. schematic (LVS) verification run.

Reporting cell-name conflicts without merging

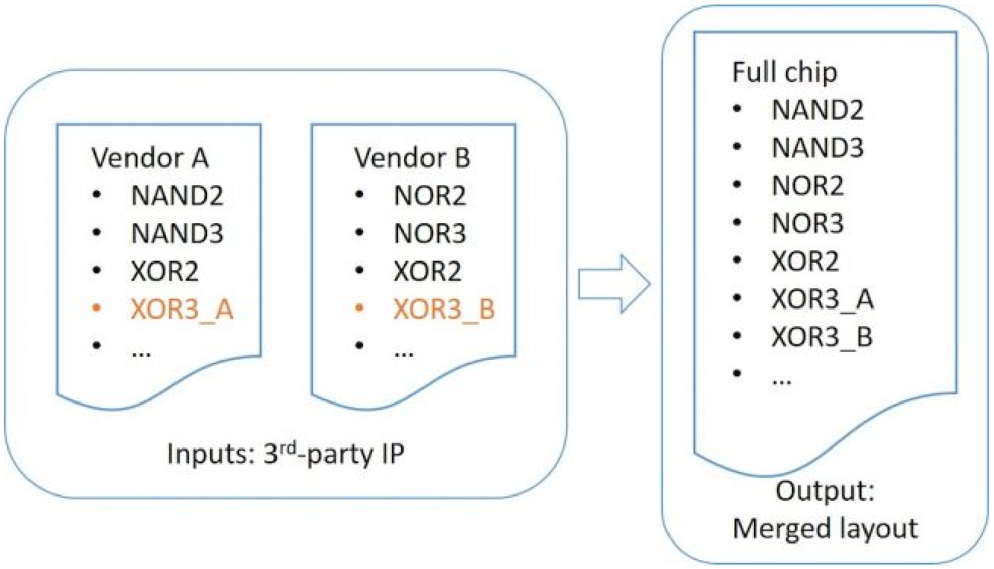
Design teams may want to review their 3rd-party IP for conflicts near the beginning of a project, when actually merging the layout may not create meaningful output. In this situation, design teams can get the conflict information they need (and save time) by using an option that instructs the FileMerge functionality to just report the list of conflicts, without actually writing the merged layout. As with the merge reporting, designers can combine this option with the smart renaming option to list only the cell name conflicts in which the contents differ (Figure 6).



(https://www.techdesignforums.com/practice/files/2020/10/Fig6_report-only-rename.jpg)

FIGURE 6 The report-only option allows designers to view any cell name conflicts without actually merging the layout

They can then rename the conflicting cells in the IP as needed, prior to the actual merging (Figure 7). This approach gives designers the information they need early in the design process and, in later stages, enables them to create a merged layout that clearly indicates the source of the content, maintains hierarchical efficiency, and minimizes file size.



(https://www.techdesignforums.com/practice/files/2020/10/Fig7_Rename-before-merge.jpg)

FIGURE 7 Identifying cell-name conflicts where the content differs prior to merging enables designers to rename the cells in the IP before any merging occurs

Conclusion

When design teams work with multiple 3rd-party IP suppliers, they will often encounter duplicate cell names across multiple IPs. These cell-name conflicts must be managed across all IP to avoid generating meaningless verification results that waste time, frustrate designers, and affect tape-out schedules. Simply renaming all cells in 3rd-party IP can cause downstream issues during physical and circuit verification. The best solution is to identify all cell-name conflicts, analyze cell contents to identify true differences, and meaningfully rename only the cell-name conflicts with content differences.

EDA solutions offer efficient, easy-to-use, automated options for identifying, renaming, and reporting these cell conflicts. Multiple renaming options let design teams create a process tailored to their needs and workflows, while multiple reporting options enable teams to obtain the information they need at every stage of the flow. Adopting an automated cell renaming process enables teams to create the clearest, most accurate, and most efficient full-chip layout in the least amount of time. The results? A clear design process, straightforward chip assembly, a clean and meaningful design hierarchy, faster EDA tool runtimes, and faster time to tape-out.

About the author

Dennis Joseph is an advanced product engineer supporting Calibre interfaces in the Design-to-Silicon division of Mentor, a Siemens business. His primary focus is the support and enhancement of the Calibre DESIGNrev layout viewer. Prior to Mentor, Dennis held positions at Intel in post-silicon validation and at Motorola in hardware test automation. He received an MS in electrical and computer engineering from the University of Florida. Dennis may be reached at dennis_joseph@mentor.com.