Leveraging Physically Aware Design-for-Test to Improve Area, Power, and Timing

By Nitin Parimi, Sr. Principal Product Engineer, Cadence Design Systems

Time-to-market pressures and the increasing need for faster and smaller devices require an integrated methodology for design-for-test (DFT) synthesis. Performing physical placement immediately after synthesis to technology gates, rather than after the entire logic and DFT synthesis process is complete, can result in significant area, power, and timing closure improvements. This white paper describes Cadence’s physically aware DFT methodology using the Cadence® Encounter® tools, and compares results using this methodology vs. more traditional design flows.

Contents

Introduction ......................................1
Why Physically Aware DFT? ..........1
Limitations of Scan-Chain Reordering During Place and Route ..........2
Physically Aware DFT Methodology ...2
Extending Physical DFT Beyond Scan ......................................3
Conclusion ........................................4

Introduction

Increased pressures on design teams to deliver faster, smaller devices in less time has required EDA companies to develop an integrated methodology to incorporate physical design information during DFT synthesis. This solution must consider the placeable area (or size) of the circuit as well as routing blockages and hard macro placement locations. It must also be able to both model the wiring interconnect given component placement locations, and build and incorporate the DFT structures into the placed design while considering its physical surroundings and wiring constraints. The physically aware DFT flow in the Cadence Encounter RTL Compiler tool addresses these challenges.

Why Physically Aware DFT?

Testability has become a critical concern for ASIC designers. DFT techniques provide a means to comprehensively test the manufactured device for quality and coverage. However, without special considerations, DFT can also introduce challenges into the ASIC design process, such as increased power consumption, functional timing issues, and excessive wiring congestion.

Scan-based structured testing is a fundamental DFT methodology in which a design’s flip-flops are replaced with special scan flip-flops containing logic targeted for improving testability. Scan flip-flops are connected serially into one or more chains and allow automatic test pattern generation (ATPG) tools to control and observe the sequential state of the design and to generate patterns to achieve the highest fault coverage.

When connecting scan chains, DFT synthesis tools can accommodate clock-domain crossings, clock-edge mixing, and voltage domain crossings by adding hardware such as lock-up latches and voltage level shifters, but satisfying all these constraints could lead to excessive additional wiring along the scan path, resulting in wiring congestion or a larger overall circuit footprint.
Limitations of Scan-Chain Reordering During Place and Route

Traditionally, scan-path wire length reduction is left as a post-processing step performed by back-end physical implementation tools, which use scan-chain reordering algorithms once the design has been placed. Although this approach can be effective, it has limitations:

- Due to scan chain reordering costs, layout tools only reorder long wires, leaving much room for scan-path wire length reduction.
- Lock-up latches, clock crossings, clock inversions, and other logic on the scan path act as “break points” across which flops cannot be reordered.

These limitations result in scan chains that are spread out over greater distances than necessary. This requires more buffering, which consumes a larger area and higher power, and makes timing closure more difficult.

Physically Aware DFT Methodology

Physical placement involves determining the exact physical coordinates of where to place each of the synthesized technology gates. As shown in Figure 1, physical placement has traditionally been performed after the entire logic and DFT synthesis process is complete. However, if we modify this methodology to place the design immediately after synthesizing it to technology gates, preferably with an input design floor plan, the placement information can optimally guide DFT logic insertion.

Most recently, we implemented both the traditional and the physically aware DFT methodology on a wireless communications chip. For the physically aware DFT implementation, we used Cadence’s Encounter Digital Implementation System to place a scan-mapped netlist, prior to inserting DFT. Then, we used Cadence’s Encounter RTL Compiler Advanced Physical to process the placement information and assign scan flops to scan chains, and finally performed further scan chain reordering in the Encounter Digital Implementation System. In the traditional flow, we simply used Encounter RTL Compiler to connect the scan chains, then took the netlist into the Encounter Digital Implementation System to reorder the chains.

Figure 2 shows the results achieved for scan flops in three different chains using both the traditional scan approach (Figure 2a) and the physically aware approach (Figure 2b). As shown in Figure 2b, scan flops in each of the physically aware chains reside in the same physical region, separate from flops in other chains, resulting in less wiring within the chains and minimal cross wiring of elements between chains when compared to physically unaware chains. Since scan flops have been allocated to the scan chains using placement locations of locally neighboring flops, further scan-data path wiring reduction achieved using back-end scan reordering tools is minimal. As shown in Figure 2a, the scan flops in the physically unaware chains are much more scattered even after using the Encounter Digital Implementation System to reorder them.
The physically aware scan methodology provides an impressive 14% reduction in scan-chain wire length, which also leads to significant improvements in timing, power, and area. See Table 1.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Traditional Scan</th>
<th>Physically Aware Scan</th>
<th>Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timing (total negative slack)</td>
<td>50.5ns</td>
<td>29.0ns</td>
<td>42.6%</td>
</tr>
<tr>
<td>Timing (worst negative slack)</td>
<td>171ps</td>
<td>145ps</td>
<td>15.2%</td>
</tr>
<tr>
<td>Area</td>
<td>65.7%</td>
<td>64.0%</td>
<td>1.7%</td>
</tr>
<tr>
<td>Scan-chain wire length, post reorder</td>
<td>1786096</td>
<td>1535628</td>
<td>16%</td>
</tr>
<tr>
<td>Total power*</td>
<td>100mW</td>
<td>95mW</td>
<td>5%</td>
</tr>
</tbody>
</table>

* Normalized average switching and internal power

Table 1: Physically aware scan results using Encounter Digital Implementation System

High drive strength buffers are often inserted to drive signals across long wires, but this additional buffering can cause higher power dissipation. The more efficient physically aware scan path wiring requires less buffering, resulting in a 5% power savings and a 1.7% overall area footprint reduction for this design. A more efficient scan path wiring can also simplify timing closure. Total negative slack (TNS), i.e., the sum of all violating paths, offers the greatest improvement because the shorter scan-path wires improve not only scan path timing, but also functional path timing, which may be difficult to close. This approach frees up routing resources that can then be used for more timing-critical paths and to reduce buffering on scan flop outputs, which drives both scan and functional paths.

Extending Physical DFT Beyond Scan

As shown in Figure 1, DFT consists of much more than scan chains. Test compression, for example, uses hardware to reduce ATPG patterns test time and test data volume. Various self-testing techniques, such as logic built-in self test (LBIST) and memory built-in self test (MBIST), can also cause significant place-and-route congestion unless managed using a similar placement-based physically aware solution.

IEEE 1149.1 boundary-scan testing is another DFT methodology that facilitates board-level interconnect testing, independent of on-chip system logic. Boundary-scan cells are inserted between I/O pads connected to each functional chip port and the system logic. The serially connected boundary-scan register provides controllability and observability to board-level interconnects.

Since JTAG boundary-scan cells directly intercept functional paths between design ports and core logic, their order in the boundary-scan shift register may result in long crossing wires along functional paths, which may lead to increased congestion.

Cadence’s Encounter RTL Compiler leverages I/O pad placement information to optimally order the boundary-scan cells in the boundary-scan register.
As shown in Figure 3, the order of the boundary-scan cells in the serially connected boundary-scan chain is decided based on the proximity of each of the connected I/O pads. This ensures that each boundary-scan cell is in the same region as the connected I/O pad and prevents long crisscrossing wires across the chip.

**Conclusion**

Cadence’s physically aware DFT solution is an efficient approach to preventing wiring congestion caused by DFT hardware. Modifying the traditional DFT synthesis flow by leveraging placement information to guide DFT insertion can result in significant improvements in area, power, and timing closure.