AC484
Application Note
Timing Closure Checklist

a Microchip company

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Learn more at www.microsemi.com.

51900484. 1.0 2/19

Contents

Figures

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 1.0

The first publication of this document.

2 Timing Closure Checklist

Timing closure in complex FPGA designs is a challenging problem to resolve. This application note provides checklist to understand the general techniques and various features available in Microsemi's Libero SoC for timing closure. As a prerequisite, the user must have a good knowledge of timing concepts and analyzing reports to improve timing closure.

In general, each phase of the FPGA design flow must be optimized so the input to the subsequent stage is optimal. The following sections describe the checklist to be considered for Design, Constraints, Synthesis, Place and Route, and Timing Verification.

2.1 References

The following documents contain further information about timing and design constraints:

- For more information about timing closure on Microsemi SmartFusion2 SoC, IGLOO2, and RTG4 FPGAs, see *WP0202: Timing Closure on Microsemi SmartFusion2 SoC, IGLOO2, and RTG4 FPGAs*.
- For more information about PolarFire timing constraints, see *UG0730: PolarFire FPGA Timing Constraints User Guide*.
- For more information about PolarFire design constraints, see *UG0776: PolarFire FPGA Design Constraints User Guide*.

2.2 Checklist For Design

For the design phase, it is important to align the user-defined RTL with the device architecture for optimizing performance.

The following checklist must be considered for the design phase:

- Understand the device architecture and plan the design architecture effectively. This also includes understanding the device hard IPs and I/O pin layout to effectively plan the design data flow and design I/Os.
- Custom or user-defined RTL must meet proper coding guidelines.
- Use synchronous design methodologies and pipeline the design if timing is not met.
- Handle cross clock domains using custom or user-defined RTL.
- For resets, use asynchronous assertion and synchronous de-assertion.
- Register hard block inputs and outputs.
  - **Example**: Select pipelined mode for LSRAMs to provide more margin than non-pipelined mode.
- If possible, minimize the size of design memory and math functions because, the fixed locations of RAM and Math IP blocks can spread the logic which impacts performance.

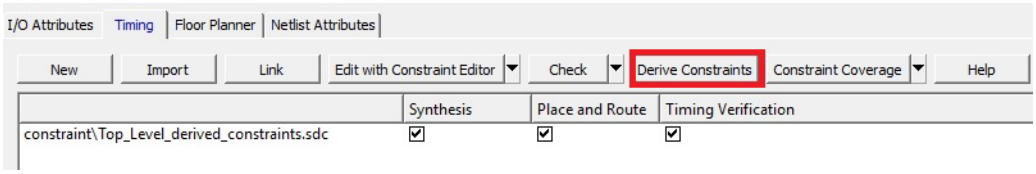
2.3 Checklist For Constraints

While applying design constraints, derive the general constraints file for the overall design and also create constraint files for user-defined clocks. These constraints are used for Synthesis, Place and Route, and Timing Verification. The Libero Constraints Manager tool enables deriving and defining various constraints and timing exceptions.

The following checklist must be considered for Constraints:

1. Generate the constraints file for the design using the Derive Constraints option as shown in Figure 1, page 3.
   Libero generates a timing SDC file based on the user configuration of IP core and components used in the design.

Figure 1 • **Derived Constraints**

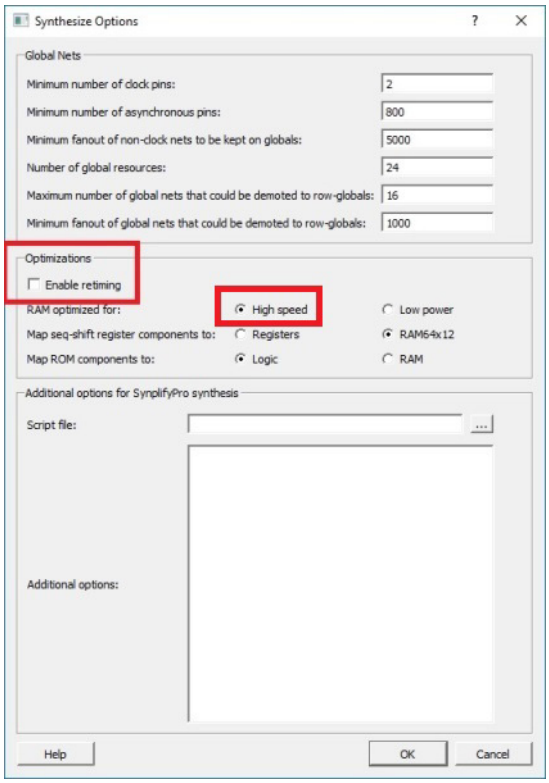


2. Examine the `Top_Level_derived_constraints.sdc` file.
3. Define the missing and generated clocks in a new file so that this information is not overwritten whenever **Derive Constraints** is run.
4. Apply proper constraints on all inputs and outputs.
5. Define timing exceptions (for example, multi-cycle path and min/max delay, and so on).
6. Use the `set_clock_groups` command for unrelated clocks in the user-defined SDC file instead of `set_false_path`.
7. If a larger die is used, consider floor planning (assigning placement constraints) your design.

2.4 Checklist For Synthesis

Ensure that the latest version of the Synplify tool is configured in Libero. The Libero release notes provides the information about the tools required and their versions. If the timing report from synthesis is not clean, you can open the Synthesis Options window and select the options available as shown in Figure 2, page 3 and re-run synthesis.

Figure 2 • **Synthesis Options**



The following checklist must be considered for Synthesis:

1. Optimize critical paths at synthesis stage using the following methods:
   - Review reports.
   - Minimize LUT levels by optimizing the HDL code.
   - Minimize high-fanout nets by replication.
   - Only clocks and resets must be considered for global net routing. Check the Global nets report to determine if any nets need to be removed from the globals.

2. Enable re-timing in synthesis options to improve timing.

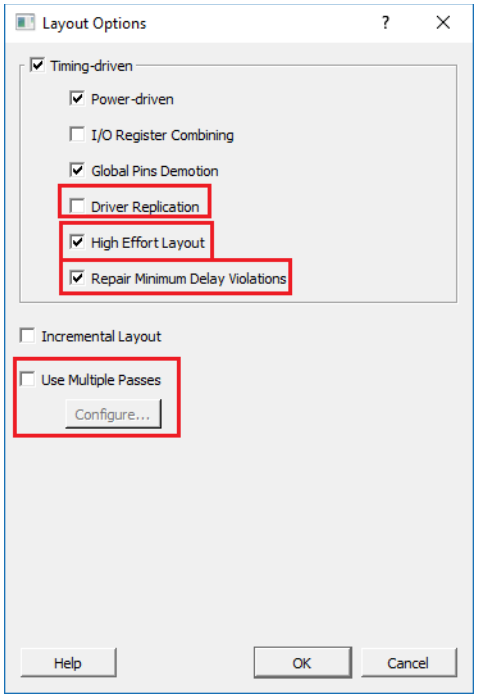
Note: Re-timing is the process of automatically moving registers (register balancing) across combinational gates to improve timing, ensuring identical logic behavior at the same time.

3. Enable RAM optimization for high-speed in the Synthesis Options window.

2.5 Checklist For Place and Route

The Libero Place and Route tool enables configuring various place and route options for an optimized performance. If the timing reports from Place and Route are not clean, you can open the Layout Options window and select the options available as shown in Figure 3, page 4.

Figure 3 • **Layout Options**



The following checklist must be considered for Place and Route:

1. Use placement constraints only when the timing is not met.
2. Generate the constraints coverage report to view the enhancement suggestions and use them to clean the report.
3. Enable I/O register combining using the Layout Options window, or using I/O PDC. I/O register combining improves pad-to-register input timing and pad-to-out output timing.

If timing violations are reported, enable the following options on the Layout Options window to improve timing:

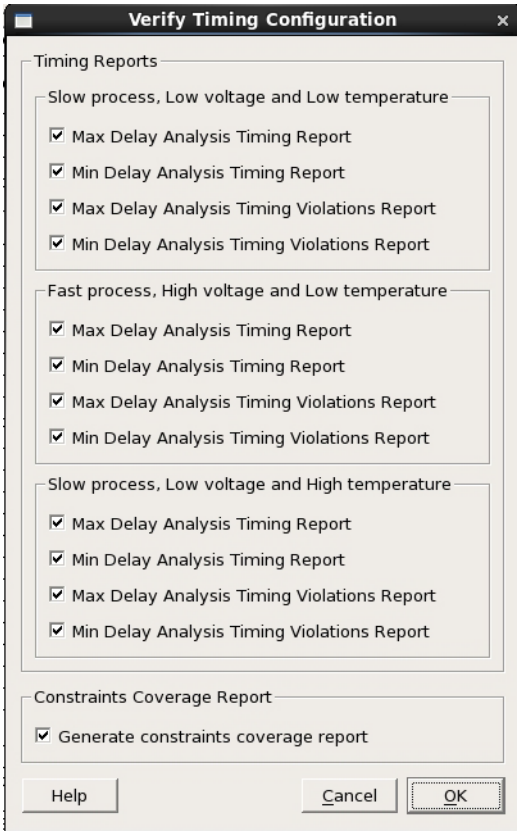
1. **High Effort Layout**—to improve the likelihood of achieving layout success.
2. **Driver Replication**—to replicate critical net drivers to reduce timing violations.
3. **Repair Minimum Delay Violations**—to repair violations due to minimum delay violations for timing-driven Place and Route mode.
4. **Multiple Pass**—to run multiple passes of Place and Route and select the best among the passes.

Note: These options increase the runtime and may improve timing.

2.6 Checklist For Timing Verification

The Libero Verify Timing tool generates timing reports, which include detailed static timing analysis and timing violations based on different process, voltage, and temperature corners. A variety of reports can be generated by selecting the options shown in Figure 4, page 5.

Figure 4 • **Verify Timing Options**



The following strategy helps analyze the timing reports and fix violations:

1. Review the constraints coverage report and ensure that the design is properly constrained.
2. Review the timing reports and understand the violating paths if timing violations are present. Start with fixing the critical paths with the largest negative slack.
3. Review the design placement using ChipPlanner and investigate suboptimal placement. Resolve placement issues by manual placement or by adding placement constraints.
4. Fix the violating paths by changing the RTL or by adding timing exception constraints as required.

2.7 Conclusion

This application note provides a general checklist and the options available to close timing. Depending on the design, either one or multiple recommendations in this document may need to be applied to close timing.