Libero SoC v12.4

Release Notes

08/2020

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

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.

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA

Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Fax: +1 (949) 215-4996

Email: sales.support@microsemi.com

www.microsemi.com

51300226-3.0/08.20

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.

Revision 3.0

- Section 5.1, Supported 64-bit Operating Systems. For Ubuntu 18.04, added that Synopsys and Mentor do not directly support the Ubuntu platform.

Revision 2.0

- Added new section 4.20.1, Installer Appears to Hang or Become Stalled.
- Section 4.12, Design with Relative Paths. Added descriptions for the following know issues:
  - Broken links are shown for locally copied files when project is migrated to a different machine
  - Missing components in SmartDesign when project is migrated from Windows to Linux and vice-versa
  - Components inside SmartDesign are not updated correctly when project is migrated between different machines
- Added Place & Route section to Known Issues.
- Added RT PolarFire CG1509 section to Known Issues.

Revision 1.0

Revision 1.0 is the first publication of this document.

Contents

Contents

1. Libero SoC v12.4 Software Release Notes

The Libero® system on chip (SoC) v12.4 unified design suite is Microchip's flagship FPGA software, for designing with Microsemi's latest power efficient flash FPGAs, SoC FPGAs, and rad-tolerant FPGAs. The suite integrates industry standard Synopsys Synplify Pro® synthesis and Mentor Graphics ModelSim® simulation with best-in-class constraints management, debug capabilities, and secure production programming support.

Use Libero SoC v12.4 for designing with Microsemi's RTG4 Rad-Tolerant FPGAs, SmartFusion®2 and IGLOO® 2® SoC FPGAs, and PolarFire FPGAs.

To design with Microsemi's older Flash FPGA families, use Libero SoC v11.9 and subsequent service packs.

To access datasheets, silicon user guides, tutorials, and application notes, visit www.microsemi.com, navigate to the relevant product family page, and click the **Documentation** tab. Development Kits & Boards are listed in the **Design Resources** tab.

Note: Libero SoC v12.4 does not support Classic Constraint Flow. IGLOO2, SmartFusion2 and RTG4 projects using the 'Classic' flow cannot be opened in this release. See Migrating an Existing Project Created with Classic Constraint Flow to Enhanced Constraint Flow for details about how to migrate Classic Constraint Flow projects to the Enhanced Constraint Flow.

1.1. Customer Notification (CN) Support

Libero SoC v12.4 includes changes that address certain important issues.

1.1.1 RTG4 SET Filter Delay Calibration Update

RTG4 devices employ Single Event Transient (SET) filters for SET mitigation. These filters are optionally available for flip-flops in Fabric LEs, IOFF, Math blocks, and SRAM blocks via Libero SoC's global SET mitigation setting or the NDC set_mitigation constraint. SET mitigation is built into the SerDes and FDDR flip-flops. SpaceWire clock and data recovery circuits embedded in the CCCs include a similar, delay-based glitch filter. Libero SoC v12.4 includes an update to ensure SET filters use the intended, Process, Voltage, and Temperature (PVT)-compensated delay values (600ps typical) from the built-in Delay Calibration (DELCAL) circuits located at the SE and NE device corners. Previous versions of Libero allowed SET filers to use an unintended filter delay value. This may result in incorrect ECC operation of LSRAMs in non-pipelined ECC mode with SET mitigation enabled.

While the updated SET delay calibration has been tested extensively, a full validation has not been completed at this time due to mandatory closure of the lab facilities during the COVID-19 pandemic. Validation will be completed as soon as public health conditions permit reopening of lab facilities.

Refer to section 2 for instructions about migrating a design to Libero SoC v12.4. For more information, refer to PCN20005.

1.1.2 RTG4 SERDES and FDDR PLL Enhanced Calibration

RTG4 PLLs can experience loss of lock at high temperature after being initialized, via device power-up or PLL reset, at cold temperature. Once loss of lock occurs, the PLL lock can be recovered by issuing a reset to the PLL. Libero SoC v12.4 enhances the calibration of the single-thread PLLs in the Fabric DDR controller PLL (FPLL) and for the SerDes PLL (SPLL) in the XAUI and PCI Express physical coding sublayer so thatthe

VCO is provided with additional gain at reset, and is capable of maintaining lock throughout the entire military temperature operating range. FDDR and SerDes configuration cores will generate enhanced CoreABC initialization microcode that applies the PLL calibration sequence during the subsystem initialization.

There is no core invalidation. Refer to section 2 for instructions about migrating a design to Libero SoC v12.4.2 Migrating Designs to Libero SoC v12.4

For more information, refer to CN19009B.

1.1.3 Configure I/O States During JTAG Programming Tool

For versions of Libero SoC, v12.0 through v12.3, using the configuration tool to "Configure I/O States During JTAG Programming" can cause a loss of user programming settings selected via "Configure Programming Options", "Configure Action/Procedure" for the "Run Programming Action" step, and "Generate Bitstream" options. This issue applies to all SmartFusion2, IGLOO2, RTG4, and PolarFire devices.

If you used the "Configure I/O States During JTAG Programming" in Libero SoC v12.0 through v12.3, you must update to Libero SoC v12.4. Refer to section 2 for instructions about migrating a design to Libero SoC v12.4.2 Migrating Designs to Libero SoC v12.4

For more information, refer to CN20006.1 and CN20003.3.

1.1.4 PolarFire FPGA TVS Powerdown

The automatic Powerdown of the TVS Sensor between conversions feature of the PolarFire (TVS) Temperature-Voltage Sensor has been found to be ineffective for its intended design purpose. For this reason, it is being deprecated from the TVS module and will no longer be supported in software.

PF_TVS v1.0.110, which is part of Libero SoC v12.4, addresses this issue. Refer to section 2.1.4 for instructions about migrating a design to use this core version.

For more information, refer to CN19029.

1.1.5 PolarFire FPGA TVS Temperature Flags

The TRIGGER_TEMP_LOW and TEMP_HIGH flags are not handled properly when the Libero SoC TVS configurator generates the TVS module. As a result, the temperature flags of the PolarFire FPGA Temperature-Voltage Sensor(TVS) trigger at wrong values when set by users in Libero SoC.

PF_TVS v1.0.110, which is part of Libero SoC v12.4, addresses this issue. Refer to section 2.1.4 for instructions about migrating a design to use this core version.

For more information, refer to CN19030.

1.1.6 PolarFire FPGA System Controller and JTAG TRSTB

In PolarFire designs created by Libero release prior to v12.4 that enable the System Controller Suspend mode, the system controller may unexpectedly stop and enter suspend mode without completing the proper device initialization. Libero SoC v12.4 implements the change to STAPL file generation, Flash Pro Express, and programming using Flash Pro 4/5/6 .

There is no design invalidation. If your design enables the System Controller Suspend mode, update the project to Libero SoC v12.4. for programming or regenerate the STAPL file.

For more information, refer to CN20003.1.

1.1.7 PolarFire Netlist Generated for LSRAM Asymmetric Two-port RAMs

PolarFire designs created by Libero release prior to v12.4 that contain Asymmetric Two-port LSRAM configurations of ratio W32:R2 incorrectly writes bits 18 and 19. Additionally, Asymmetric Two-port LSRAM configurations of ratio W32:R1 incorrectly writes bits 17 through 19. Libero SoC v12.4 corrects the Write-data bits of these configurations.

There is no design invalidation. If your design contains either of these configurations, you must update the project to Libero SoC v12.4. Rerun the Synthesize/Compile tool, configure the P&R Incremental option, and complete the entire flow.

For more information, refer to CN20003.2.

1.1.8 PolarFire FPGA PCIE Register Update

The default values in the PCIe BAR register generated prior to Libero SoC v12.4 were incorrect and could impact the BAR addressing in their PCIE system. Incorrect default register values related to PCIE_BRIDGE in the PolarFire Register Map documentation has been updated in Libero SoC v12.4 to correctly match silicon.

There is no core invalidation. If your design contains a PCIe component with configured PCIe BAR register using the Libero SoC default values prior to v12.4, update the project to Libero SoC v12.4 and upgrade to PCI Express core v2.0.104.

For more information, refer to CN20003.4.

1.1.9 PolarFire Certain Unused IOs were not Tristated when Programmed

PolarFire programming files generated by Libero release prior to v12.4 could lead certain I/Os to being driven, even if they were not used in the design. Libero SoC v12.4 corrects the attributes of these specific unused I/Os. The impacted pins are listed in the device PPAT tables.

CCC_NW_PLL0_OUT[0:1]	CCC_NW_PLL1_OUT[0:1]
CCC_NE_PLL0_OUT[0:1]	CCC_NE_PLL1_OUT[0:1]
CCC_SE_PLL0_OUT[0:1]	CCC_SE_PLL1_OUT[0:1]
CCC_SW_PLL0_OUT[0:1]	CCC_SW_PLL1_OUT[0:1]

There is no design invalidation. If your design didn't use any of the above I/Os, you must update the project to Libero SoC v12.4. Rerun the "Generate FPGA programming data" tool and generate the programming file.

For more information, refer to CN20003.5.

1.1.10 PolarFire LSRAM CLK-to-OUT Delays

During timing characterization over temperature, an adjustment was made to the timing model of the PolarFire LSRAM CLK-to-OUT delays that impacts lower temperatures delays (i.e., industrial and extended):

- For max clock-to-out delays, only Bit 0 is impacted: Internal regression shows that less than 1% of designs are impacted and, when there is an impact on design performance, it is less than 2%.
- For min clock-to-out delays, all bits are impacted: In this case, the delays through the LSRAM are large enough not to show up as min delay critical paths. Internal regression confirmed that no violation was observed due to this change.

There is no design invalidation. Although it is unlikely that this change will have any timing violations on existing designs, we recommend rerunning static timing analysis with SmartTime using Libero SoC v12.4 to ensure your design remains timing violation clear.

For more information, refer to CN20003.6.

1.2. New Device Support

1.2.1 RT PolarFire

Libero SoC v12.4 introduces the Radiation Tolerant **RT PolarFire** device **RTPF500T-CG1509** MIL temp, STD speed-grade. SynplifyPro can infer Soft triple module redundancy (TMR) for inferred FFs. You can use TMR to mitigate SEUs in any part of the design. P&R separates the TMR FFs physically to mitigate clock transients and upsets.

You can start making pin assignments for the new CG1509 Ceramic package. To optimize PCB layout and signal integrity of DDR signals, DDR3 and DDR4 pin assignments will be altered in the next release of Libero SoC. These changes will affect only DDR3 and DDR4 I/Os, and will not affect any other I/O assignment. Place and route performed with Libero SoC v12.4 will be invalidated when Libero SoC v12.5 is invoked. Timing and Power is in "Advance" state. Programming and BSDL export of this device is not yet enabled.

1.2.2 PolarFire Military Operating Condition

Production timing support has been added for the following devices:

- MPF200TS MIL temp, 1.0V, STD speedgrade
- MPF500TS MIL temp, 1.0V, STD speedgrade

Preliminary timing support has been added for the followingdevices:

- MPF200TS MIL temp, 1.05V STD speedgrade
- MPF300TS MIL temp, 1.05V STD speedgrade
- MPF500TS MIL temp, 1.05V STD speedgrade

1.3. Software Features and Enhancements

1.3.1 Runtime Improvements

Libero SoC v12.4 includes faster timing report generation (up to 40% compared to v12.3), particularly on large designs with multiple clock domains.

Synthesis runtime of PolarFire designs improve by an average of 25% due to Automatic Compile Points being enabled by default.

1.3.2 PolarFire Post Layout Simulation

Libero SoC v12.4 enables Back-Annotated simulation for fabric components in PolarFire designs.

1.3.3 Post Layout Clock Domain Crossing (CDC) Report and GUI

Libero SoC v12.4 introduces a clock domain crossing report that is generated after P&R. This report identifies potential areas of data loss or meta-stability that can be caused by signals crossing clock domains. To view this report from the SmartTime GUI, select the Tool → Reports → Report CDC menu.

For more details, refer to the SmartTime User Guide.

1.3.4 Timing Reports

Libero SoC v12.4 extends the configurations of Verify Timing for customizing the generated Timing reports in PolarFire designs. The new Report settings:

- Are available in XML, Text, CSV (Tcl only), and HTML (Tcl only) formats.
- Limit the number of reported paths per section.
- Limit the number of expanded paths per section.
- Limit the number of parallel paths per expanded path.
- Maximum slack threshold for Timing Violations report.

In Libero SoC v12.3, we introduced consolidated "multi-corner" timing and violation reports (see section **1.3.5** of Libero SoC v12.3 release notes for a detailed description). Libero SoC v12.4 removes the frequency information from the Summary section of the consolidated "multi-corner" timing report. This information was an outdated inheritance from the single-corner timing report and was misleading because its value did not account for CDC requirements. This information still appears in the various single-corner reports for compatibility reasons.

1.3.5 Minimum Delay Violations Repair Enhancement

In Libero SoC v12.4, min-delay repair has been enhanced to explore worst-case paths from all corners simultaneously. This results in fewer iterations to close timing for different corners.

1.3.6 Synthesis Enhancements

SynplifyPro 2019.03MSp1-1 included in Libero SoC v12.4 contains the following enhancements:

- Ternary operations inference using a single carry-chain – comparator (up to 66)
- PolarFire LSRAM Inference with asymmetric widths
- PolarFire Improved runtime: Automatic compile points enabled by default. SynplifyPro executes multi-threaded when Compile points are enabled. Design iterations resynthesize only the Compile points that were modified, reducing the synthesis time.

For more details, refer to the following documents:

- Synopsys FPGA Synthesis Synplify Pro ME P2019.03MSP1-1 User Guide
- Synopsys FPGA Synthesis Synplify Pro ME P2019.03MSP1-1 Reference Manual
- Synopsys FPGA Synthesis Synplify Pro ME-P2019.03MSP1-1 HDL Language Support Reference Manual
- Synopsys FPGA Synthesis Synplify Pro ME P2019.03MSP1-1 Command Reference Manual
- Synopsys FPGA Synthesis Synplify Pro ME P2019.03MSP1-1 Attribute Reference Manual
- Synopsys® FPGA Design Microsemi Edition Release Notes

1.3.7 Relative Path Enhancement for Linked Files

Support for linked files has been enhanced in Libero SoC v12.4 to make it easier to port projects across users and/or machines. Now users can specify an environment variable that defines the base path from which the link file relative path must be calculated. If the environment variable is not specified, the path calculation defaults to legacy behavior.

For more details, refer to the Libero SoC User Guide.

1.3.8 Multiuser Environment

Libero SoC v12.4 provides the ability to lock the core vault location, the synthesis and simulation profiles to the software release. One can create a special profile in the software release installed area to define which vault, synthesis and simulation tools to use for that release. Once defined in that release-level profile, these settings cannot be overridden from Libero. This guaranteed inadvertent changes to the settings for users working with multiple projects and releases of Libero and Libero SoC.

To create an admin profile, invoke adminProfile.exe from the Libero or Designer's /bin location.

For more details, refer to the UG0758: User Guide PolarFire FPGA Design Flow Libero SoC v12.4 or the UG0691 User Guide Libero SoC Design Flow Libero SoC v12.4 - SmartFusion2, IGLOO2, RTG4.

1.3.9 Debug Enhancements

Identify P2019.03MSp1-1 included in Libero SoC v12.4 supports debugging using FlashPro6.

1.3.10 Libero Installer Upgrade

For ease of use, the Libero SoC v12.4 installer has been enhanced to include an optional Libero license installation flow. The Libero installer now supports three integrated flows:

- Libero tool installation
- Libero license installation
- Integrated tool and license installation

The installer includes a Tcl-based example that runs the Libero-synthesis-simulation flow to check for license issues.

1.3.11 Ubuntu Support

Libero SoC v12.4 introduces support for Ubuntu 18.04. Customers can now install Libero SoC v12.4 on an Ubuntu platform. For details, refer to "System Requirements" on page 47. For installation details, refer to the UG0710 Libero SoC Linux Environment Setup User Guide.

1.3.12 2TB and Greater Partition Size Support

Libero SoC v12.4 now fully supports Linux environments using partitions greater than 2TB, including installing Libero and locating Libero projects.

1.4. New Silicon Features and Enhancements

1.4.1 PolarFire

Transceiver Enhancements

Libero SoC v12.4 enhances the PolarFire Transceiver Enhanced Receiver Management (ERM) Solution to support independent receive and transmit data rates in PMA and PCS-8b10b modes on the same physical lane.

Each PolarFire Transceiver Enhanced Receiver Management (ERM) component has the following options: full duplex, independent receive and transmit, and receive only. These options allow functionality to be assigned efficiently to each lane and help solutions that use small package offerings when transceiver lanes are limited.

Libero SoC v12.4 adds a new option to configure Transmit PLL in integer mode. The Transmit PLL supports Fractional-N and a new integer mode. Fractional-N provides fine grain flexibility for reference clock selection. The new integer mode provides designers with a simplified calculation for direct integer multiples of the reference clock.

Libero SoC v12.4 enhances the Transmit PLL Jitter Attenuation solution significantly when using the Custom Protocol mode. The enhanced solver allows the JA PLL to achieve close to or 0ppm with any valid FIN clock frequency. When using the recovered clock from a Transceiver lane as reference clock source (FIN) to the Transmit PLL using 'Fabric routing', the input jitter on JA reference clock will increase due to noise effects from the FPGA fabric routing. If possible, new designs should avoid using Fabric clock source to the JA_PLL and use the Dedicated clock source instead.

Libero SoC v12.4 SmartDebug for PolarFire simplifies transceiver receiver signal integrity analysis by including an eye mask that specifies the minimum eye opening.

The PolarFire Transceiver Solution supports an expanded set of CTLE settings. Note that these settings are not yet fully validated, and should not be considered Production.

- For A2 - Short/Medium in CDR, CDR Auto and DFE modes the following RX_CTLE options are available:
    - No_Peak_+9.22dB
    - No_Peak_+4.53dB
    - No_Peak_+1.76dB
    - 5GHz_+3.14dB
- For A1 - Short/Medium in DFE mode the following RX_CTLE options are available:
    - No_Peak_+11.10dB
    - No_Peak_+6.13dB
    - No_Peak_+3.39dB
    - 6GHz_+2.73dB
    - 6GHz_+3.12dB

For details about CTLE settings, refer to AC483: PolarFire FPGA Transceiver Signal Integrity Application Note.

IOD Interfaces Enhancements

Previous Libero releases limited the number of I/Os per interface to 32. Libero SoC v12.4 removes this limitation and allows up to 128 I/Os per interface.

Libero SoC v12.4 also adds HS_IO_CLK to RX/TX_CLK_G training for both the receiver and transmit generic IOD interfaces which, combined with HS_IO_CLK to DQ training, enables maximum data rate for the IOD interfaces

For more details, refer to the IO User Guide.

CCC Configuration Report

Libero SoC v12.4 generates a configuration report for the CCC containing the final values of the CCC configuration registers. This report is generated in the “Design and Memory Initialization” Libero stage, in

.xml and .txt formats, and viewable in the report view.

PolarFire SRAM (AHBLite and AXI) ECC Support

Libero SoC v12.4 adds ECC support for the PolarFire SRAM (AHBLite and AXI) core. If the SRAM is initialized with data at power-up, the ECC data is also initialized properly after power-up.

PolarFire XCVR Sourced Fabric Clocks and Jitter Compensation

The PolarFire XCVR can source three different clocks into the fabric, TX_CLK, RX_CLK, and the REFCLK (FAB_REF_CLK). These clocks will contain high frequency jitter that is not reported by Libero in the timing report and SmartTime. It is recommended that users add clock uncertainty constraints to these clocks in their design.

The following list is recommended values for clock uncertainty per clock, resource, and speed-grade.

- FAB_REF_CLK on Global: 275ps for STD, 200ps for –1
- FAB_REF_CLK on Regional: Not supported
- TX_CLK_G on Global: 300ps for STD, 225ps for –1
- TX_CLK_R on Regional: 225ps for STD, 150ps for –1

- RX_CLK_G on Global: 325ps for STD, 250ps for –1
- RX_CLK_R on Regional: 250ps for STD, 175ps for –1

Below is an example clock uncertainty constraint. This constraint would be added to the user's timing SDC file.

```
# TX_CLK and RX_CLK on Regionals

set_clock_uncertainty -setup 0.150 [ get_pins { PF_XCVR_ERM_LANE2/I_XCVR/LANE0/TX_CLK_R } ]
set_clock_uncertainty -setup 0.175 [ get_pins { PF_XCVR_ERM_LANE2/I_XCVR/LANE0/RX_CLK_R } ] # TX_CLK and
RX_CLK on Globals

set_clock_uncertainty -setup 0.300 [ get_pins { PF_XCVR_ERM_LANE2/I_XCVR/LANE0/TX_CLK_G } ]
set_clock_uncertainty -setup 0.325 [ get_pins { PF_XCVR_ERM_LANE2/I_XCVR/LANE0/RX_CLK_G } ] # FAB_REF_CLK
on Global

set_clock_uncertainty -setup 0.275 [get_clocks PF_DDR4_C0_0/CCC_0/pll_inst_0/OUT1]
```

A future release of Libero will add these constraints automatically to the derived constraints from the Constraints Manager.

1.4.2 RTG4

RTG4 SRAM (AHBLite and AXI) ECC Support

Libero SoC v12.4 adds ECC support for the RTG4 SRAM (AHBLite and AXI) core. The option to Initialize RAM for simulation when ECC is enabled will be added in a future release.

2 Migrating Designs to Libero SoC v12.4

2.1 Design and Core Invalidation

2.1.1 Programming Files Invalidation

As a result of CN20006.1 and CN20003.3, Libero SoC v12.4, for all families, will automatically update the affected programming settings and invalidate bitstream generation tools accordingly. The impacted tools are “Configure Programming Options”, “Configure Action/Procedure” for the “Run Programming Action” step, and “Generate Bitstream Options”.

Rerun any of the affected tools as needed, and reprogram device with updated bitstream.

2.1.2 RTG4 Filter Calibration Design Invalidation and RTG4UPROM Core Update

As a result of PCN20005, all RTG4 designs with completed programming file are invalidated upon migrating a project created in a Libero release prior to v12.4 to Libero SoC v12.4. Completed designs that do not use the uPROM will not require re-running Place & Route, unless that step is invalidated due to other required updates, per the Libero SoC log window.

In the uPROM, total number of available 36-bit words has been reduced from 10,400 to 10,370. This allows space to be reserved for the instructions required to calibrate the filter delay.

New designs are not affected by this upgrade because they will use the updated RTG4UPROM core.

However, completed designs containing uPROM instance require a uPROM core version upgrade and revert to a pre-synthesis/pre-compile state.

Existing designs are impacted as described below:

- If the design does not contain a uPROM component, Libero SoC v12.4 will not invalidate Place and Route, but will invalidate “Generate FPGA Array Data”, “Generate Bitstream”, “Export Bitstream”, and “Export FlashPro Express Job”. The impacted flows must be rerun.

- If the design contains a uPROM component, Libero SoC v12.4 will invalidate “Synthesize” / “Compile”:

    a. See section 2.3 for information about how to update a core. Upgrade the uPROM component to v2.1.100 and rerun the entire flow. The run will fail if the clients exceed 10,370 words.

    b. For a netlist-only project, regenerate the uPROM and rerun synthesis outside Libero.

Rerun the tool flow, including “Generate FPGA Array Data” and “Generate Bitstream” steps. Reprogram device with updated bitstream.

2.1.3 RTG4 SERDES and Fabric DDR Core Update

As a result of CN19009B, the enhanced PLL calibration cores for FDDR PLLs and SerDes PLLs are included in the Libero SoC v12.4 software release and must be integrated into existing designs for the enhanced calibration to be applied to FDDR PLL and SerDes SPLL. With this update, all single-thread RTG4 PLLs will have lock stability independent of junction temperature rise during operation within datasheet limits.

See section 8.0 (Appendix B) for details on the enhanced PLL solution for FDDR and SerDes PLL.

See section 2.3 for information about how to update a core.

2.1.4 PolarFire Temperature and Voltage (TVS) Interface Core Invalidation and Update

The PolarFire Temperature and Voltage (TVS) Interface core has been updated in Libero SoC v12.4 to account for the changes described in CN19029 and CN19030. If a design contains the TVS core, and:

- Temperature sensing channel is enabled: Libero SoC v12.4 will invalidate “Synthesize”. The TVS core must be upgraded to the latest version and the full flow must be rerun.
- Temperature sensing channel is not enabled. Libero SoC v12.4 will not invalidate “Synthesize”.

See section 2.3 for information about how to update a core.

2.2 Core Enhancements and Upgrades

If a project created in a Libero release prior to v12.4 contains the following cores, and the cores have been generated, they do not need to be upgraded after migrating the project to Libero SoC v12.4. However, if the core needs to be generated again for any reason (for example, a change in parameters), the latest version from the Catalog must be downloaded and used.

Libero SoC v12.4 includes the following **PolarFire** core updates from Libero SoC v12.3.

Display Name	Version	Change Description
PolarFire DDR3	2.4.111	There are no functional changes for this version. This new version uses the latest training and IOD sub-core versions.
PolarFire DDR4	2.4.111	There are no functional changes for this version. This new version uses the latest training and IOD sub-core versions.
PolarFire LPDDR3	2.3.111	There are no functional changes for this version. This new version uses the latest training and IOD sub-core versions. The latest training sub-core version fixes the fast simulation mode for low data rates.
PolarFire QDR	1.7.100	There are no functional changes for this version. The unused QVALID port has been removed from the core.
Transceiver Interface	3.0.102	The transceiver interface core now supports the 'Enhanced Receiver Management' option for the 'Tx and Rx (Independent)' mode.
PCI Express	2.0.104	Some default register values have been updated to match the actual hardware values. See section 1.1.8 for more information.
Transmit PLL	2.0.202	The Transmit PLL has been enhanced to improve the Jitter Attenuation (JA) integer solution. It also improves the custom solution to achieve close to or 0ppm resolution.
PolarFire IOD CDR Clocking	2.1.104	This core has been enhanced to display the TX_CLK_G clock frequency.

Display Name	Version	Change Description
PolarFire IOD Generic Receive Interfaces	2.0.123	The IOD Generic Receive Interfaces core offers multiple enhancements: Up to 128 I/Os per interface (from 32 in previous version). HS_IO_CLK to RX/TX_CLK_G training, which combined with HS_IO_CLK to DQ training enables maximum data rate for the IOD interfaces.
PolarFire IOD Generic Transmit Interfaces	2.0.108	The IOD Generic Transmit Interfaces core offers multiple enhancements: Up to 128 I/Os per interface (from 32 in previous version) HS_IO_CLK to RX/TX_CLK_G training, which combined with HS_IO_CLK to DQ training enables maximum data rate for the IOD interfaces. Specialized clock port to be connected to the new PolarFire IOD Generic Transmit Interfaces Clocking core required to complete the HS_IO_CLK to RX/TX_CLK_G training solution.
PolarFire IOD Generic Transmit Interfaces Clocking	1.0.121	The PolarFire IOD Generic Transmit Interfaces Clocking is a new core that is needed as part of implementing the HS_IO_CLK to RX/TX_CLK_G training solution for the PolarFire IOD Generic Transmit Interface.
PolarFire RGMII to GMII	1.2.111	There are no functional changes for this version. This new version uses the latest IOD sub-core version.
PolarFire SRAM (AHBLite and AXI)	1.2.105	The new version of this core supports ECC
PolarFire Dynamic Reconfiguration Interface	1.0.102	There are no functional changes for this version. This new version has an updated User Guide reference.
PolarFire Initialization Monitor	2.0.105	The CALIB_START port which was added in Libero SoC v12.3 has been removed in this release. The CALIB_START port is automatically removed when upgrading to this new version.
Temperature and Voltage Sensor Interface	1.0.110	This version removes the deprecated POWERDOWN option. It also fixes the computation of the temperature thresholds. See section 1.1.4 and section 1.1.5 for more information.
CoreSmartBERT	2.6.101	There are no functional changes for this version. This new version uses the latest transceiver core version.

Libero SoC v12.4 includes the following **RTG4** core updates from Libero SoC v12.3.

Display Name	Version	Changes Description
RTG4 SRAM (AHBLite and AXI)	1.0.110	The new version of this core supports ECC
RTG4 uPROM	2.1.100	Makes room for SET delay re-calibration. All RTG4 designs with RTG4UPROM are invalidated to pre-synthesis/compile state. Users MUST upgrade to 2.1.100. See section 1.1.1 for more information.
RTG4 DDR Memory Controller	2.0.100	This core version generates a new CoreABC initialization sequence that implements the FDDR FPLL temperature drift calibration solution. See section 1.1.2 for more information.
RTG4 DDR Memory Controller with Initialization	2.0.100	This core version generates a new CoreABC initialization sequence that implements the FDDR FPLL temperature drift calibration solution. See section 1.1.2 for more information.
RTG4 High Speed Serial Interface (PCIe, EPCS & XAUI)	2.0.100	This core version generates a new CoreABC initialization sequence that implements the Transceiver SPLL temperature drift calibration solution. See section 1.1.2 for more information.
RTG4 High Speed Serial Interface 1 – EPCS and XAUI – with Initialization	2.0.100	This core version generates a new CoreABC initialization sequence that implements the Transceiver SPLL temperature drift calibration solution. See section 1.1.2 for more information.
RTG4 High Speed Serial Interface (EPCS & XAUI)	2.0.100	This core version generates a new CoreABC initialization sequence that implements the Transceiver SPLL temperature drift calibration solution. See section 1.1.2 for more information.
RTG4 High Speed Serial Interface 2 – EPCS and XAUI – with Initialization	2.0.100	This core version generates a new CoreABC initialization sequence that implements the Transceiver SPLL temperature drift calibration solution. See section 1.1.2 for more information.

See section 2.3 for information about how to update a core.

2.3 Core Update Procedure

Perform the following procedure to update a core version:

- Download the latest version of the core into your vault.
- Upgrade each configured core in your design to the latest version by right-clicking on the core component in the design hierarchy and selecting **Replace Component Version**.
- Regenerate the design.
- Derive the Timing Constraints again from the Constraint Manager tool to use the latest generated core constraints.
- Rerun the tool flow.

3 Resolved Issues

The following table lists the customer-reported defects and enhancement requests resolved in Libero SoC v12.4. Resolution of previously reported “Known Issues and Limitations” are also noted in this table.

Case Number	Description
493642-2701742107	Change in timing values after running Verify Timing in v12.3.
493642-2625748880	Clock latency value for IO-Reg[External Hold analysis] paths.
493642-2689492312	In SmartTime, right-clicking on a timing path or expanded timing path does not highlight the net in ChipPlanner.
c493642-2673418401	Cross probing unstable with zoom in feature
493642-2706645305	IO reg combining not working in Netlist flow Resolution: Libero SoC v12.4 fixes an issue with the organization of Netlist Design Constraint (NDC) files for the EDIF/VM post-synthesis netlist flow.
493642-2718476813 493642-2718624799	Libero v12.3 doesn't honor NDC file for EDIF/VM flow Resolution: Libero SoC v12.4 fixes an issue with the organization of Netlist Design Constraint (NDC) files for the EDIF/VM post-synthesis netlist flow.
493642-2629904414	POWERDOWN checkbox in PolarFire TVS IP configurator must be removed Resolution: The Temperature and Voltage Sensor (TVS) Interface core version 1.0.110 fixes this issue.
493642-2676058215	PF_TVS: temp_low & temp_high flags not correctly working Resolution: The Temperature and Voltage Sensor (TVS) Interface core version 1.0.110 fixes this issue.
493642-2672542990	PF_QDR: QDR_PF illegal pad issue in Libero 12.2 Resolution: The PolarFire QDR core version 1.7.100 fixes this issue.

Case Number	Description
493642-2638786872	PF_SRAM_AHBL_AXI: Enhancement request for EDAC Resolution: The PolarFire SRAM (AHBLite and AXI) core version 1.2.105 implements this enhancement.
493642-2671317816	G4_DDR: For igloo2 DDR2 type and PHY width of 8, burst length of 4 is not supported. Resolution: • Burst Length 8 (BL8) is supported DDR3, DD2 and LPDDR and all data widths. • Burst Length 4 (BL4) is supported only for DDR2 and LPDDR and only for data width of 32. • Burst Length 16 (BL16) is supported only for LPDDR and only for data width of 16. • Design Rule have been added for non-supported configurations
493642-2677814246	Enhancement request - Add AXI4 streaming BIF. Resolution: The AXI4 Bus Interface (BIF) has been added to Libero SoC v12.4 and can be used when creating a core from an HDL module.
493642-2709015110	PF: Export IBIS deletes the P&R flow in VM netlist design. Resolutions: This defect has been resolved in Libero SoC v12.4.
493642-2625770311	Enhancement Request: TCL command to report timing closure pass/fail based on the results of Verify Timing command. Resolution: The following messages are displayed in the log window at the end of the Verify Timing command: If timing violations: Error: Timing constraints have not been met. If no timing violations: Info: Timing constraints have been met. If no timing constraints were set: Warning: No Timing constraints are set.

Case Number	Description
493642-1986842740 493642-2326709729 493642-2531788673 493642-2552248549	RELATIVE_PATH: Libero SoC prj file has the absolute path for linked HDL Resolution: See the Relative Path Enhancement for Linked Files section for details about this new feature.
493642-2711545965	Inconsistency in exporting stp for MPF devices in Libero v12.3 Resolution: When running the export_bitstream_file command, all file formats specified by the -format option are properly as expected.
493642-2690544739 493642-2728908419	STP and .dat files not getting exported using exported tcl in Libero 12.3 Resolution: When running the export_bitstream_file command, all file formats specified by the -format option are properly as expected.
493642-2436656358 493642-2432126472 493642-2523772273 493642-2609521421 493642-2615450581	Relative paths support in Libero Resolution: See the Relative Path Enhancement for Linked Files section for details about this new feature.
493642-2544111080	FP6: Add EXTEST2 instruction to CM3 code to solve I/O glitch issue.
493642-2695957773 493642-2727763417	PF_SOC: Issues Power Breakdown by voltage rails/Resource in "MSS & MDDR" sheet
493642-2722049359	P&R crash without any message Resolution: This defect has been fixed in Libero SoC v12.4.
493642-2725028410	G5layout.exe crash in Libero SoC 12.3 Resolution: This defect has been fixed in Libero SoC v12.4
493642-2651035974	Enhancement request for VERIFY TIMING when running multi-pass P&R Resolution: When multi-pass P&R is run, Verify Timing is now properly cleaned-up. To obtain the timing reports, explicitly call Verify Timing after P&R.

Case Number	Description
493642-2698404809	RTG4: Add check for valid combination of ulock bits at bitstream gen level.
493642-2643973119	Enhancement request: Save for list of signals in Live Probe.
493642-2615986097	RTG4 LVDS Output IBIS Models (software export).
493642-2678703755, 493642-2705388493, 493642-2707071569	VHDL Simulation issue in Modelsim ME (standard version), which was released with Libero SoC v12.2.
493642-2596989658	Static ODT "OFF" should not allow to set ODT impedance value
493642-2664199123	RTG4 design fails min timing on slow corner. Resolution: Repair min-delay violations analyzes paths from all corners starting with Libero SoC v12.4.
493642-2700092229	new_project tcl command has confusing error message. Resolution: When specifying the family name in the new_project Tcl command, make sure to enter the proper case (e.g., SmartFusion2 and not SMARTFUSION2). The error message has been fixed to show the list of valid families with proper case.
493642-2695913352	TCL command to identify Libero version Resolution: You can use 'libero –version (-release)' on the command line to get the Libero SoC version (release) value. You can use Libero SoC Tcl command get_libero_version (get_libero_release) to get the Libero SoC version (release) value.
493642-2644630854	project_archive TCL command missing Resolution: The project_archive Tcl command has been added to the Libero SoC v12.4 TCL command reference guide.
493642-2689522793	PF_UPROM: TCL: Issue adding the clients to the PF_UPROM core through create_and_configure_core TCL command.

Case Number	Description
493642-2635269681	PF_DRI is not documented Resolution: The PolarFire Dynamic Reconfiguration Interface core version 1.0.102 has a proper reference to the User Guide for this feature.
493642-2689492312	Not pointing the nets path in Netlist Viewer- Flat Post-Compile Resolution: The net selection feature has been fixed in Libero SoC v12.4.
493642-2636657018	Post Layout Simulation support in Libero PolarFire Resolution: See the PolarFire Post Layout Simulation section for details about this new feature introduced in v12.4 for PolarFire.
493642-2677619049	CCC glitch filter with clock inversion is not recognized in STA.
	Overlap of instance names in SmartTime path viewer.
493642-2302984395	The node names in the Smart time are overlapping with each other.
493642-1738180702	Advanced Type: record is not supported for HDL+ flow (Create Core from HDL) Resolution: Explanation message: Error: Failed to create Core from HDL as ports with advanced data types are present in file
	Synplify crashes with Compiler Error [range.c:466 Expecting constrained record field].
493642-2595223516	Synthesis Error - Internal Error in c_ver.exe.
493642-2689541559	RTG4: CCC/CCCDYN have wrong of INIT string size. Resolution: The RTG4 simulation file rtg4.v has been updated to match the INIT string size.
493642-2636978831 493642-2637115919 493642-2673908851	RTG4 CCC: CCC producing output clocks and asserting lock when no input clock is present.

Case Number	Description
493642-1445497887 493642-1480712716 493642-1819489087 493642-2219798281 493642-2260829519 493642-2687217793 493642-911572233 493642-2670019886 493642-2361822794 493642-2692514691	VHDL and HDLPLUS support for Complex types. Resolution details: Libero SoC v12.4 enforces proper checks for all port and generic types that are not supported when instantiating an HDL module into a SmartDesign or when creating a HDL core. The documentation has been enhanced to clearly describe which types are supported and also provides example of how to create wrappers that convert the unsupported types to the fundamental types that are supported.
493642-2728200443	PolarFire CCC: Export register information in a table. Resolution: The PolarFire CCC now generates information about registers in CCC.
493642-2646702704	PolarFire IOD Generic Receive Interface: When choosing one of the Dynamic Data Alignment templates, and then opening the Advanced tab, the tool does not let the user select "Expose dynamic delay control". Resolution: The entire Expose Dynamic Delay Control option has been greyed out.
493642-2698320125	TCL command 'exit' has been added.
493642-2657278778	Compiler generated "net <net_name> does not have a driver" warnings in compile report generated.

4 Known Issues and Limitations

4.1 Catalog Cores

4.1.1 Core Generation Language

- PolarFire SgCore and SystemBuilder cores generate only Verilog files, regardless of the preferred HDL language selected in the Libero project.
- VHDL users desiring to simulate designs containing these cores must use mixed-language simulation (available with ModelSim ME Pro, which is bundled with this release, and requires a Gold, Platinum, or Eval license).

4.1.2 Linux: Core Generation Fails in Batch Mode when the DISPLAY Variable is Not Set

The following Direct Cores cannot be generated in Libero using batch mode via Tcl when the DISPLAY variable is not set on a Linux machine:

CoreAXI4SRAM	CoreCIC	CoreCordic	CoreABC
CoreEDAC	CoreDDS	CoreFIFO	CoreFFT
CoreFIR_PF	CoreRSDEC	CoreRSENC	

4.1.3 Core Version Upgrade

When upgrading the core version for components created for PolarFire System Builder cores from an older version to the latest, both core versions must download to the vault. Otherwise, the core version upgrade fails.

4.1.4 New Cores Are Available Pop-up Message

New cores are available pop-up appears when opening a project. If the cores are not present in the catalog, ignore the popup.

4.1.5 Core Generation Issues

The Generation of Licensed Cores Core APB3 and Core ABC fails if performed through Remote Desktop.

4.2 Project Manager Design Hierarchy

4.3.1 HDL Language Duplicate Modules

- If a design has duplicates between a core module and an HDL module, opening the project does not show the Design Hierarchy properly. To fix this issue, build the Design Hierarchy after opening the project.
- If a design has duplicates between the elaborated modules of the core and a normal HDL module, the modules are not shown as duplicate modules in the Design Hierarchy.
- If two different VHDL files have the same signature (same inputs, outputs, and architecture), the modules are not detected as duplicate modules in the Design Hierarchy.

4.2.2 Multiple Definitions Error Message for RTG4FCCC and RTG4FCCCECALIB

When RTG4FCCC and RTG4FCCCECALIB are used in same design, the following multiple definitions error message is reported in the log window. This message does not stop the user design flow.

```
Error: The CCCAPB module is defined in multiple files. Duplicate modules are not
supported.
```

Workaround: Select any ccc_comps.v file, as the files have the same content.

4.3 Tcl Support Limitations

4.3.1 Parameters for SgCore and SystemBuilder Components are not Documented

To configure these cores using Tcl:

1. Use the GUI to configure the core as desired.
2. Export the core configuration Tcl description by selecting the “Export Component Description(Tcl)” action on the right-click menu of the component in the Design Hierarchy.
3. Use the exported Tcl command to create the configured core in a regular Tcl script.

Note: The following set of cores cannot be configured using Tcl. As result, the Export Component Description (Tcl) option is not supported:

- SmartFusion2/IGLOO2 MSS/HPMS component
- SmartFusion2/IGLOO2 System Builder component
- RTG4 DDR memory controller with initialization (RTG4FDDRC_INIT)
- RTG4 High Speed Serial Interface 2 – EPCS and XAUI – with Initialization (NPSS_SERDES_IF_INIT)
- RTG4 High Speed Serial Interface 1 – EPCS and XAUI – with Initialization (PCIE_SERDES_IF_INIT)

4.4 SmartDesign

4.4.1 Modify Memory Map Feature Should Not be Used

Do not use the Modify Memory Map action in the Libero SoC v12.4 release to connect peripherals to buses in the SmartDesign canvas. Otherwise, Libero may crash or produce an incorrect or incomplete memory map. Instead, connect peripherals to bus slave positions manually, as per the desired memory map.

4.4.2 Export Component Description (Tcl)

When the Export Component Description (Tcl) command is executed on a SmartDesign, pin groups created by Libero might convert to Tcl commands and the exported Tcl script might errors-out when executed.

Workaround:

- Delete the converted line(s) that have create_pin_group commands that are failing from the exported file.
- Delete the created SmartDesign.
- Re-execute the Tcl script.

4.4.3 Special Characters in Element Names

SmartDesign does not support special characters for instance, module and port names - e.g., module

\hdl1\ with port \port1\- . The following actions will fail:

- Instantiation of HDL module may fail.
- Promotion to a top-level port of an instance pin.
- Un-doing the deletion of an instance with special characters in the name.
- The generated netlist may be incorrect.

4.5 Synthesis

4.5.1 SynplifyPro Mapping of Sequential Shift to uSRAM Does Not Support Initial Values

PolarFire devices do not support initial values on registers for Sequential-shift constructs mapped to uSRAMs. If an initial value is specified for a register in RTL, Synplify ignores the value and issues a warning.

.

4.5.2 Standalone Synthesis Flow

Libero SoC v12.4 users can synthesize designs outside the Libero SoC software using Synopsys SynplifyPro directly. When using this flow, the following additional steps are necessary to synthesize and implement a design:

- For Windows, make sure the <install location>/Designer/data/aPA5M/polarfire_syn_comps.v is added as a source file to the SynplifyPro project. This file contains module declarations with timing

information for PolarFire primitives not known to Synopsys.

- For Linux, make sure the <install location>/Libero/data/aPA5M/polarfire_syn_comps.v is added as a source file to the SynplifyPro project. This file contains module declarations with timing information for PolarFire primitives not known to Synopsys.
- Many configured cores generate timing constraints. For optimal results, make sure these constraint files are passed to synthesis. These constraint files must also be imported into Libero along with the synthesis gate level netlist for optimal place, route, and timing analysis results. Core-generated constraint files must be modified so that constraints are expressed using the proper hierarchical name of the configured cores in the top-level design.

4.5.3 SmartFusion2/IGLOO2 with Libero Derived Constraints May Cause Synthesis Warning

SmartFusion2/IGLOO2 with Libero derived constraints can generate warnings during synthesis. These warnings are issued because Synplify Pro optimizes unused logics, and can be ignored.

4.6 I/O Editor

4.6.1 DRC Validation

The DRC check in the I/O Editor does not validate all the constraints set in the tool. You must run Place and Route to validate these constraints.

4.7 Netlist Viewer

4.7.1 Multiple Views

Opening two or more views (for example, Hierarchical RTL View and Flattened Post-Compile View) in Netlist Viewer may cause a crash due to memory usage. Avoid opening multiple views for large designs.

4.8 Place and Route

4.8.1 Place & Route Failure for "get_clocks" in the "-to" Portion of Timing Constraint

In Libero SoC v12.4, designs with a false path constraint which uses a "get_clocks" in the "-to" portion of the path may, in rare cases, crash during placement. If this happens, consider replacing the false path constraint with an asynchronous clock group constraint, or replace the "get_clocks" syntax with a specific set of pins. Another alternative is to create a Place & Route specific SDC file in the Manage Constraints tool where this constraint is commented out.

4.9 PolarFire Block Flow

4.9.1 Block Components

Libero SoC v12.4 supports Block Flow. Only Fabric components (LUT, SLE, RAM, MATH) may be instantiated in a block. All other components (PLL, DLL, XCVR, DDR, IOD) must be part of the top-level design and cannot be instantiated in blocks.

4.9.2 Replication

Replication option cannot be enabled in a design containing blocks.

```
Info: The design contains instantiated blocks. Driver Replication is not
supported in this mode and was turned off.
```

4.10 PolarFire SSN Analyzer

4.10.1 Deviation

For all PolarFire MPF300T/TS/TL/TLS -FCG1152 devices, SSN Analyzer simulated data deviates from the Silicon measured data. This deviation can range from 20% to 60%.

4.11 SmartPower

4.11.1 Initialize Frequencies & Probabilities with Vectorless Analysis

In SmartPower, selecting the **Tools -> Init Freqs & Prob** menu crashes the tool if the “Use vectorless analysis“ option is selected.

Workaround: Perform power analysis providing the Frequency/Toggle rate directly in SmartPower or through an .sdc file.

4.12 Design with Relative Paths

4.12.1 ENVM Serialization Client

SmartFusion2/IGLOO2 ENVM Serialization Client not working correctly when a path relative to the environment variable path is used.

Description: Create a project in SF2/IGLOO2 with Relative Path Settings. While configuring eNVM in MSS/System Builder, click on add Serialization Client. With "Use Relative Path" option, try to link any mem file relative to the environment variable path. This causes "Maximum Devices to program" option to have a red mark next to it, which prevents any further action to link the serialization client.

Workaround: Link the mem file with absolute path or relative path to the project instead.

4.12.2 Broken Links are Shown for Locally Copied Files When Project Migrates to a Different Machine

This behavior was observed when:

- A Libero project with HDL source/stimulus files linked with a relative path using the ENV variable and the HDL modules instantiated in a SmartDesign component moves from one machine to another machine.
- The directory path set in the ENV variable for linked files is different on the two machines.
- Some linked files have been locally copied to the project before moving to the target machine.

Workaround: Use the **Change link** option to change the link of these files.

4.12.3 Components are Missing in SD When Project Migrates from Windows to Linux and Vice-Versa

This behavior was observed when a Libero project with HDL source/stimulus files linked with a relative path using the ENV variable and HDL modules instantiated in a SmartDesign component is moved from one Windows to Linux or from Linux to Windows. For example, when a project created on Windows is opened on Linux, the components inside the SmartDesign canvas are missing, even though the files and modules under **Design Hierarchy** tab are shown correctly as expected. The same behavior occurred with Linux-to-Windows migrations.

Workaround: Delete the missing HDL module instances from the SmartDesign canvas, re-instantiate them, and then generate the SmartDesign.

4.12.4 Components Inside SD are not Updated Properly When Project Migrates between Different Machines

This issue was observed when a Libero project with HDL source/stimulus files linked with a relative path using the ENV variable and HDL modules instantiated in a SmartDesign component moves from one machine (Machine 1) to another (Machine 2) in the following scenario:

1. The directory path set in the ENV variable for linked files is different on the two machines, and
2. The linked files are not present in the directory path set in the ENV variable of the target machine.

Previously, when the project was opened on Machine 2, the linked files were shown with broken links in the Design Hierarchy, and the HDL module instances inside the SmartDesign canvas were shown as missing. With this release, updating the broken links for the files to a location where the files are stored on Machine 2 restores the links correctly in the Design Hierarchy; however, the HDL module instances inside SmartDesign canvas are still shown as missing.

Workaround: Delete the missing HDL module instances from the SmartDesign canvas, re-instantiate them, and then generate the SmartDesign.

4.13 PolarFire Silicon Support Limitations

4.13.1 Transceivers

PCIe

- During BFM simulation of the PCIe AXI interface (master or slave), the simulator may print warning messages about AHB signals, such as “HRESP”. The warning messages can be ignored.
- Collapsible group settings in the PF_PCIE configurator may not be clearly visible when invoked on laptop screens. You may collapse all groups except the one you want to view or configure for a better experience.

Transceiver Reference Clock

- Enabling on-die-termination and external VREF on the Transceiver Interface Reference Clock I/O is not supported in the I/O editor. However, these options can be set in the I/O PDC file.

  Refer to UG0715: PDC Commands User Guide (PolarFire) for more information.
- The connection from the Transceiver Interface Reference Clock I/O to the South-East PLL for all the reference clocks associated with Transceiver Interface Quad 0,2 and 4 lanes is not available in the software. Place and Route will fail if this connection is attempted.

Transceiver Reference Clock Placement

The XCVR_REF_CLK instance does not show up and cannot be placed using XCVR view when only FAB_REF_CLK output is being used.

Workaround: If XCVR_REF_CLK is instantiated in a user design and the dedicated clock output is not used to drive Transceiver or PLL (only fabric output is used), use PDC to place this XCVR_REF_CLK instance

ERM is Not Available for MPF300XT or ES Devices

Do not use the Enhanced Receiver Management option in the XCVR_ERM (Transceiver Interface) core with the MPF300XT or MPF300T/TS_ES devices. This restriction is not currently enforced by the Libero software.

4.13.2 IOD Interfaces

PolarFire IOD Generic Receive Interface Data Rate

In the PF_IOD_RX configurator, duplicate Data rate DRC check messages appear for the presets RX_DDRX_B_G_FA_HSIO, RX_DDRX_B_G_DYN_HSIO, and RX_DDRX_B_R_DYN_HSIO. Ignore the second DRC message. Correct preset data rates are 700, 1600, and 500, respectively.

PolarFire IOD Generic Receive Interfaces Simulation Failures at Some Data Rates

In fractional aligned or fractional dynamic, simulation may fail for some data rate. The output of the PLL will either stop toggling or keep changing frequency. There’s no workaround.

PolarFire IOD Generic Transmit Interfaces Forwarded Clock Option

On selecting Ratio = 1, 'No forwarded clock option' under clock to data relationship is not allowed in PF_IOD_GENERIC_TX configurator.

Workaround: Instead of using PF_IOD_GENERIC_TX configurator, use PF_IO Configurator to perform simple DDR_IN, DDR_OUT, and DDR_INOUT operations. Users can achieve the same functionality as TX ratio 1 without a forwarded clock.

4.13.3 PF_CCC DLL Reference Clock Minimum Frequency

If the PolarFire CCC is configured in DLL or PLL-DLL cascaded mode, the DLL reference frequency must be greater or equal to 133 MHz. When opening a design using the custom flow, this rule is not enforced.

Users should make sure they generate CCC configurations that do not violate this requirement. CCC configurations generated with Libero SoC v12.1 or later meet this requirement.

4.13.4 PF_SPI

PF_SPI Macro does not support SPI-Slave mode. It supports SPI-Master mode only. The SS_OE and CLK_OE ports should always be tied high. Libero SoC v12.4 will error out if these two ports are not properly tied high.

4.13.5 I/O’s: SSTL15 On-Die Termination Values Are Programmed Incorrectly

If the ODT value for an SSTL15 I/O is selected as 20 Ohm or 30 Ohm for MPF300XT/TES/TSES devices, an incorrect setting is programmed

Workaround: Do not use 20 or 30 Ohm on-die termination values for the affected devices.

4.14 PolarFire Power Estimator

4.14.1 Design without I/Os

A PolarFire device with no IOs that is kept active using 1.8V supply draws current roughly in the range of 10mA@25C, which is not reported by the power tools.

4.15 PolarFire Post Layout Simulation

4.15.1 Exception Error Related to Power Nets in Some BA Designs

There may be an exception in PolarFire designs when "Generate Back Annotated Files" is run for post-layout simulation. Although Libero generates the exception, ba.v/ba.vhd files are generated for all three corners and can run simulations successfully.

4.16 RT PolarFire Silicon Support Limitations

4.16.1 CG1509 Pinout

To optimize PCB layout and signal integrity of DDR signal, the assignment of DDR3 and DDR4 pins in Libero SoC v12.4 will change in Libero SoC v12.5. These changes will affect only DDR3 and DDR4 I/Os, and will not affect any other I/O assignment. Place and route performed with Libero SoC v12.4 will be invalidated when Libero SoC v12.5 is invoked.

4.17 RTG4 SRAM (AHBLite and AXI)

4.17.1 Initialize RAM for Simulation

RTG4_SRAM_AHBL_AXI core does not support the option "Initialize RAM for Simulation" if ECC is enabled. (Pipelined or Non-pipelined)

4.18 Identify Debugger

4.18.1 Identify Instrumentor May Hang on Windows 10 Machines

When the Identify Instrumentor is opened in Integrated mode on certain machines, the tool opens, but then freezes upon interaction. This is a rare occurrence in Windows 10 OS configurations.

Workaround: Use the Standalone Identify Instrumentor.

4.19 Aldec Simulator

4.19.1 Simulation Stuck with Aldec Simulators

PCIe BFM simulation gets stuck after one write-read BFM command while using PF_SRAM_AHBL_AXI core in Active HDL and Riviera Pro Aldec simulators.

Workaround: If you use Aldec Simulators, change the generated CoreAXI4SRAM_MAINCTRL.v file manually, as shown on the following page.

At line number #373, comment/remove below lines in starting of always block

//wr_req_gnt =1'b0;

//rd_req_gnt =1'b0;

And explicitly set for each state:

In REQ_IDLE_ST state:

wr_req_gnt = 1’b0; rd_req_gnt = 1’b0;

In REQ_WR_ST state:

wr_req_gnt = 1’b1; rd_req_gnt = 1’b0;

In REQ_RD_ST state:

wr_req_gnt = 1’b0; rd_req_gnt = 1’b1;

In default state:

wr_req_gnt = 1’b0; rd_req_gnt = 1’b0;

4.20 Installation and System Limitations

4.20.1 Installer Appears to Hang or Become Stalled

It may take long to run the Libero SoC v12.4 web installer over slower internet connections, giving the impression that the installer has stalled or is hung.

In fact, the installer is downloading files in the background. We recommend that users wait for the installation process to complete or, for faster installation times, install Download Libero SoC v12.4 installer (Windows), which is 6.7 GB.

4.20.2 FlashPro6 Driver Reinstallation Reports Error Message

Customers with FlashPro6 drivers previously installed on their system may see the following message at the end of the installation:

```
The installation of Program Debug Tool v12.4 is finished, but some errors occurred
during the install. Please see the installation log for details.
```

Resolution: Uninstall existing FlashPro6 drivers and restart the system before installing Libero SoC v12.4. If the software is already installed, ignore the above message if installation logs do not report any errors.

4.20.3 4K and 8K screens are Not Supported

4K and 8K screens are not supported in the Libero SoC v12.4 release.

4.20.4 Installation on Local Drive Only

This release is for installation on a local drive only. The Installer might report permission rights problems if the release is installed across a networked drive.

4.20.5 Visual C++ Redistributable Installation Error

On some machines, the installer may display the following message:

```
The installation of Program Debug Tool v12.4 is finished, but some errors occurred
during the install. Please see the installation log for details.
```

The above error message is harmless. If it is seen, click **Yes** and Libero SoC v12.4 will be installed successfully.

4.20.6 Installation on Windows 7

During Libero SoC v12.3 installation on Windows 7 machines, you may see pop-up warning messages about shortcuts toward the end of installation process.

You can safely ignore these messages. Click OK to close the pop-up windows and have the installation proceed and complete as expected. All Windows shortcuts will appear correctly.

4.20.7 Installation Fails when There is Insufficient Space

- In Libero SoC v12.4, the web installer quits without any notification or error message to users if there is insufficient space for the installation. In addition, the estimated space for the installation is incorrect – it reads as approximately 236MB required. Make sure there is at least 20GB free space on the target hard drive before invoking the installer.

  Note: The DVD installer will not proceed if there is insufficient space.

- Linux installer fails the first time as IATEMPDIR is not set.
- Windows Standalone Installer: Spaces in Extraction Path
  - During installation of the standalone (DVD) version, the folder to which the zip file is extracted must not contain spaces. If spaces are present, invocation of the installer will fail with the error "Windows cannot find '<truncated path to extracted folder>'. Make sure you typed the name correctly, and then try again". Rename and/or move the extracted folder to one without spaces (in the entire path).

4.20.8 Linux Package Notes

- In Libero SoC v12.4, the script bin/check_linux_req/check_linux_req.sh reports incorrectly that the Linux package xz.i686 is required for RHEL/CentOS 7.x. Package xz.i686 is not required. The correct required packages are xz-libs.x86_64 and xz-libs.i686.

- If the installer does not boot in graphical mode, additional X window system libraries might be required. For RHEL/CentOS, the following system package is recommended:

  $ sudo yum install -y libXau libX11 libXi libxcb libXext libXtst libXrender

4.21 Antivirus Software Interaction

- Many antivirus and Host-based Intrusion Prevention System (HIPS) tools flag executables and prevent them from running. To avoid this problem, modify your security setting by adding exceptions for specific executables. This is configured in the antivirus tool. Contact the tool provider for assistance.

- Many users run Libero SoC PolarFire successfully with no modification to their antivirus software. Microchip is aware of issues for some antivirus tool settings that occur when using Symantec, McAfee, Avira, Sophos, and Avast tools. The combination of operating system, antivirus tool version, and security settings all contribute to the end result. Depending on the environment, the operation of Libero SoC v12.4, ModelSim ME and/or Synplify Pro ME may or may not be affected.
- All public releases of Libero software are tested with several antivirus tools before they are released to ensure that they are not infected. In addition, the Microchip software development and testing environment is also protected by antivirus tools and other security measures.

4.22 Programming

4.22.1 Libero Programming

- Updating the security or sNVM with a security-only bitstream or sNVM-only bitstream on a device that has the Fabric programmed disables the Fabric. If the Fabric is disabled, you must reprogram the Fabric to enable it.

  Workaround:

  1. sNVM only bitstreams: Field-update bitstream files should always program the Fabric with sNVM.
  2. Security only bitstreams: Security-only bitstream should be used on a blank device only.

- If a device is programmed with a blank Silicon Signature field, it will not get erased.

  Workaround:

  1. Specify a Silicon Signature that is not blank and program the device to change the value.
  2. Perform the Erase program action to erase it.

- A SPI file that contains a Silicon Signature setting (set in Configure Programming Options) cannot be imported as a SPI bitstream file for a Recovery/Golden client (in the SPI Flash configurator).

  Workaround: Leave the optional Silicon Signature Field in “Configure Programming Options” as blank before generate SPI bitstream. If Recovery bitstream needs to include silicon signature, Use Libero SoC v12.0 software.

- If the Silicon Signature that is part of the security segment is unspecified and the security is not programmed, the previous value of Silicon Signature is retained.

- FlashPro Express does not support SPI-Flash only programming in a JTAG chain with SmartFusion2/IGLOO2 and PolarFire devices.

- Serialization feature does not work for SmartFusion2 and IGLOO2 starting from Libero SoC v12.0.

- The action device_info shows programming file type as SVF instead of PPD when job-ppd is used.

4.22.2 SPI Flash Programming

Supported Micron SPI Flash Memory Devices

This release supports only the following Micron SPI Flash memory devices:

- Using FlashPro5: MT25QL01G only
- Using FlashPro6: all members of N25Q device family.

  Note: Contact Microchip Technical Support about support for Flash memory devices from other vendors and device families using FlashPro6.
- Using FlashPro6: all members of MT25Q device family except for devices with 128MBits and 64Bits density. The issue related to 128MBits and 64Bits density devices will be fixed in Libero SoC v12.6.

  Workaround: User can program the SPI-Flash using fabric design.

SPI Flash Programming Limitations

- This tool erases the SPI Flash prior to programming. It is recommended to program the SPI Flash with Programming and Debug Tools v12.4 prior to programming other data on the SPI Flash using non-Libero programming solutions.
- Partial update of the SPI Flash is currently not supported.
- Avoid large gaps between clients in the SPI Flash, since gaps are currently programmed with 1's and increase programming times.

See the appendix for programming tables.

4.22.3 sNVM Write Fails Due to ROM Client Created by Previous Design

- If a PolarFire device is programmed with a design with an sNVM client, and then reprogrammed with a different design without an sNVM client, the sNVM client is not erased when programming with the second design is completed. If there are sNVM pages that are locked, writes to those pages will fail.
- There is no programming action to erase the sNVM completely.

  Workaround: Create a mock (dummy) sNVM client (filled with 0's) in the second design. The issue will be fixed in the upcoming release.

4.22.4 Verify During PROGRAM Action Fails for Design with Custom User Security

For Libero SoC v12.0 and above that use PolarFire designs with Custom user security options, enabling the DO_VERIFY optional procedure in PROGRAM action and executing PROGRAM action in Libero (via Run PROGRAM action) fails with "Invalid/Corrupted encryption key".

Workaround: Run the standalone VERIFY action after PROGRAM separately if needed.

4.22.5 STAPL Programming is Not Supported Using FP6

FlashPro6 does not support STAPL programming.

Workaround: Generate FlashPro Express jobs with a PPD file to enable programming for FlashPro6. Note that the PPD file is also supported by FlashPro 4 and 5.

4.22.6 Export Bitstream tcl Command Fails if "for_ihp" is 0

The “export_bitstream_file” command fails when it sets "for_ihp" parameter to 0 and unsets DAT format. Use the parameter "for_ihp" to set DAT and STAPL formats to true only when its value is "1" and should be ignored if its value "0".

Workaround: Remove "for_ihp" parameter or set its value to "1".

4.22.7 Verify Digest May Not Work with FlashPro6

“Verify Digest” action causes intermittent failure with “FP6 connection failed” and “Failed to disable FP6 programmer” error messages, that require FlashPro6 to be unplugged and re-plugged from the USB port to continue.

Workaround: Use VERIFY action to verify that the device is programmed with the expected design.

4.23 Secure Production Programming Solution

4.23.1 Job Manager May Crash During the Import Job Status Operation

Running the Import Job Status operation may cause Job Manager to crash. This can occur with designs where the User ECC Key mode is enabled and targeting SmartFusion2 or IGLOO2 M2S/M2GL060, 090, or 150 devices.

4.23.2 Job Manager Crashes When Opening an Existing Job Manager Project from v11.9

Job Manager v12.4 does not support Job Manager project files created with releases prior to v12.0.

4.23.3 Job Manager Does Not Support PolarFire DAT Export

PolarFire DAT file bitstream export from Job Manager is not supported in Libero SoC v12.3.

4.23.4 SmartFusion2/IGLOO2: eNVM Update Protection with FlashLock is No Longer Supported

Due to a silicon limitation, eNVM update protection with FlashLock has been defeatured. If a JDC file generated with a pre-v12.0 version of Libero SoC has the eNVM set to be protected by passkey, it must be regenerated with Libero SoC v12.3 without eNVM FlashLock Protection enabled. eNVM update protection continues to be provided by User Encryption Keys (UEK1, UEK2 or UEK3).

4.23.5 ERASE Action Failure for FlashPro Express Job

If a HSM FlashPro Express job has tickets for PROGRAM and ERASE actions without a ticket for the VERIFY action, the ERASE action fails. To run the ERASE action successfully, make sure a ticket for the VERIFY action is included.

4.23.6 Job Manager init_bitstream Tcl Command Limitation

If you run non-HSM flow using Job Manager on a Windows PC, without specifying the "enable_passkey_export" option in the init_bitstream Tcl command, the exported bitstream files may include passkeys.

Workaround: Explicitly set the "enable_passkey_export" option to either TRUE or FALSE in the init_bitstream Tcl command to export the correct bitstream files.

4.24 SmartDebug

4.24.1 General SmartDebug Limitations

- Initializing RAM blocks with random values in the Design Initialization Data and Memory tool will result in SmartDebug displaying incorrect values for zeroed memory blocks.

- The logical view cannot be reconstructed for:
  - LSRAM/uSRAM for port widths of x1 inferred through RTL.
  - LSRAM/uSRAM configurations when a single net of an output bus is used and others are unused (i.e., A_DOUT[0]/B_DOUT[0] for DPSRAM/uSRAM and RD[0] for TPSRAM). In this scenario, memories can be read/write using physical view.
  - LSRAM/uSRAM configurations inferred using CoreAHBLtoAXI (Verilog flow) and CoreFIFO (Verilog and VHDL flow).
  - HDL modules inferring RAM blocks that are instantiated in SmartDesign.
  - TPSRAM with ECC enabled.

  Workaround: None

4.24.2 PolarFire Transceiver Support Limitations

Plot Eye introduces a burst of errors in data traffic on Transceiver Interface lanes when started for lanes configured in CDR mode. This will be fixed in an upcoming Libero SoC PolarFire release.

Workaround: Before starting traffic, enable Eye Monitor using the Power On Eye Monitor option. This powers on the DFE and EM receivers in CDR mode, and no spurious errors will be seen during eye plot.

4.24.3 Eye Monitor Limitation on 8b10b PCS-PMA Mode XCVR Lanes

CVR lanes configured as 8b10b PCS-PMA with receiver in DFE mode, requires specific interface sequence to be followed for Eye Monitor to work.

Workaround: Perform the following sequence of steps to obtain the expected eye output:

1. Assert PCS RX RESET.
2. Optimize Receiver.
3. Plot Eye.
4. De-assert PCS RX RESET.

4.24.4 SmartBERT Limitations

- The SmartBERT IP does not work when lanes are configured at 250Mbps data rate.
- SmartBERT IP PRBS tests using PRBS patterns from the SmartBERT IP take more time to start/stop/inject error on RHEL 7.x and Cent OS 7.x platforms than on RHEL 6.x and Windows OSes.
- SmartBERT IP PRBS tests do not work when the first Transceiver lane uses an internal pattern (PRBS from XCVR PMA) and the following lane uses a SmartBERT pattern (PRBS from SmartBERT IP).

  Workaround: If multiple lanes are selected to run the tests, test patterns on every lane should be either SmartBERT IP pattern or on-chip PMA pattern.

4.24.5 Static Pattern Transmit Limitation

During the Static Pattern Transmit operation, the Receiver PLL (RX PLL) does not lock to the max run length pattern when looped back from TX to RX.

4.24.6 Eye Monitor TCL Limitation

The Power ON eye monitor Tcl command (eye_monitor_power) does not work correctly in Programming and Debug Tools v12.3 and v12.4. The Receive PLL does not lock to the incoming data after this Tcl command is run. This will be fixed in an upcoming Libero SoC PolarFire release.

PCIe Debug Limitations

- Designs using PCIe1 controller:
- RXPLL lock status is not shown in the Configuration Report UI of Debug XCVR.
- Signal Integrity parameters are shown in the Configuration Report UI of Debug XCVR.

Designs using Dual PCIe (i.e., PCIe0 and PCIe1):

- PCIe debug is not supported for designs using dual PCIe controllers.

4.24.7 PolarFire FPGA Hardware Breakpoint (FHB) Limitations

- Soft reset behavior is not consistent when the DUT is clocked at frequencies less than 160MHz. This means the device might not respond to a soft reset operation initiated from the FHB UI.

  Workaround: Perform the following steps:

  1. Halt the DUT via Live Probe.
  2. Use the FHB UI to initiate a soft reset operation.
  3. Halt the DUT again via Live Probe.

- TCL command for frequency_meter fails when run in SmartDebug. This will be fixed in an upcoming Libero SoC PolarFire release.

- Event Counter value is read as 2 for static signals (signals connected to GND/VCC) when assigned to Live Probe Channel A.

4.24.8 SmartFusion2/IGLOO2 FPGA Hardware Breakpoint (FHB) Limitations

- When the selected live probe trigger signal is zero/ground, disarming the trigger leads to the forced halt of Device Under Test (DUT).

  Workaround: Use the reset option in FHB to bring FHB to a default state and un-halt the DUT.

- Event Counter value is read as 2 for static signals (signals connected to GND/VCC) when assigned to Live Probe Channel A.

4.24.9 RTG4 FPGA Hardware Breakpoint (FHB) Limitations

- Synthesis fails when FHB auto instantiation is enabled on designs containing FCCC modules using instantiation flow (i.e., FCCC modules instantiated in RTL files).
- Live Probe channel assignment using a static signal connected to GND halts the DUT. If this occurs, initiate a soft reset operation using the FHB UI to restart the DUT.
- Halting a clock domain driven by a CCC also halts all four clock domains of the CCCs in the design. This is a silicon limitation.
- FHB is not supported for cascaded CCCs (CCCs that rely on other CCC outputs to source the reference/feedback clocks). FHB is auto instantiated, but the PLAY/HALT/STEP operations do not work.
- LiveProbe is not Retained on PROBE_READ_DATA Pin for 2nd SmartDebug Run Using FlashPro5 for RTG4 Devices.
- If the LiveProbe is set initially in the first SmartDebug session/run, it is not retained when a new session is invoked.

  Workaround: A def variable is provided "SMARTDEBUG_RTG4_FLASHPRO5_DISABLE_RESET" a value '0' is by default. A value '1' has to be overwritten to avoid reset.

4.24.10 RTG4 LSRAM Data Corruption

LSRAM data corruption occurs when performing a read to LSRAM configured in 512x36 mode through SmartDebug on the active address location.

4.24.11 Standalone SmartDebug Limitations

Microchip devices present in a chain with non-Microchip devices cannot be debugged using standalone SmartDebug. In addition, the ID code of Microchip devices cannot be read in this scenario.

Workaround: Use SmartDebug through Libero to perform these operations.

4.24.12 Auto-Chain Construct Limitation in Standalone SmartDebug

Programming fails for all device families when a standalone SmartDebug project is created using the "Construct Chain Automatically" option, and a DDC file is imported in the Programming Connectivity and Interface dialog.

Workaround: Perform one of the following steps:

1. Close and reopen the Programming Connectivity and Interface UI after importing the DDC file contents in Programming Connectivity and Interface, and then click **Run Program Action**.
2. Create a project by importing the DDC file without Auto-construct.

4.24.13 TCK Frequency Not Reflected in SmartDebug Operations

If TCK frequency is set in the Programmer Settings in Standalone SmartDebug, it is not reflected in the SmartDebug operations for SmartFusion2, IGLOO2, RTG4 and PolarFire devices.

Workaround: Set TCK frequency; program the device and then use SmartDebug features to debug. This will be fixed in an upcoming Libero SoC PolarFire release.

4.24.14 FlashPro6 Programmer Not Detected When Creating a SmartDebug Project

When SmartDebug is invoked for the first time after the installation of Program and Debug software, FlashPro6 programmer is not detected during SmartDebug project creation.

Workaround: Exit the SmartDebug instance and invoke a new instance of SmartDebug for project creation to be successful.

4.24.15 Incorrect Programmer Information Reported for FlashPro6

In Design and Programmer Information UI, Programmer type is reported as FlashPRO 3 for FlashPRO 6 in SmartDebug main window. This will be fixed in an upcoming Libero SoC PolarFire release. The issue does not exist in Device Status.

5 System Requirements

The Libero SoC v12.4 release has the following system requirements.

5.1 Supported 64-bit Operating Systems

- Windows 7 or Windows 10 OS
- RHEL 6.6-6.11 and RHEL 7.2-7.6
- CentOS 6.6-6.11 and CentOS 7.2-7.6
- Ubuntu 18.04 (Synopsys and Mentor do not directly support the Ubuntu platform)
  Note: Setup instructions for using Libero SoC v12.4 on Red Hat Enterprise Linux OS, CentOS, or Ubuntu are available in UG0710 Libero SoC Linux Environment Setup User Guide. As noted in that document, installation now includes running a shell script (bin/check_linux_req.sh) to confirm the presence of all required runtime packages.

5.2 Random-Access Memory (RAM) Requirements

Minimum of 16 GB RAM

6 Download Libero SoC v12.4 Software

The following are available for download:

- Libero SoC v12.4 for Linux
- Libero SoC v12.4 for Windows
- Mega Vault (Linux)
- Mega Vault (Windows)

Note: Installation requires administrative privileges.

After a successful installation, clicking **Help-> About Libero** shows version: 12.900.0.16.

7 Appendix A: Sample Programming and SmartDebug Times Using FlashPro5/FlashPro6

The tables in this appendix show sample programming times using FlashPro5 and FlashPro6 programmers.

The following table shows sample PPD programming times.

Devices[1]	PPD Programming Time[2] (mm:ss)		
	FlashPro5	FlashPro6	
	TCK=4MHz	TCK=4MHz	TCK=20MHz[3]
	USB 2.0	USB 2.0/3.0	USB 2.0/3.0
M2S/A2GL 005			
M2S/A2GL 010			
M2S/A2GL 025			
M2S/A2GL 050	2min 9sec	2min 10sec	2min 2sec
M2S/A2GL 060			
M2S/A2GL 090			
M2S/A2GL 150	4min 21sec	4min 19sec	3min 54sec
RTG4	2min 10sec	1min 56sec	1min 33sec
MPF100	39sec	28sec	23sec
MPF200	1min 3sec	43sec	28sec
MPF300	1min 33sec	1min 4sec	43sec
MPF500	1min 57sec	1min 34sec	1min

NOTES:

[1] FlashPro6 supports JTAG programming for all SmartFusion2, IGLOO2, RTG4 and PolarFire devices.

[2] To benefit from the improved programming time using FlashPro6, use the PPD file format for SmartFusion2, IGLOO2 and PolarFire devices. Programming time speed up with PPD will be added in future releases.

[3] To ensure successful programming at 20MHz TCK, appropriate steps need to be taken to ensure signal integrity of the JTAG signals.

The following table shows sample SPI Flash programming times (all using PPD flow)

SPI Flash Programming

Splash Kit

(N25Q00AA13GSF40G / MT25QL01GBBB8ESF-0SIT TR)[1] 10MByte data	PPD Programming Time				
	FlashPro5		FlashPro6[2]		
	TCK = 4MHZ	TCK = 15MHz[3]	TCK = 4MHZ	TCK = 15MHz[3]	TCK = 20MHz[3]
	USB 2.0	USB 2.0	USB 2.0/3.0	USB 2.0/3.0	USB 2.0/3.0
Program SPI Flash	8min 30sec	5min 29sec	22min 12sec	13min 9sec	12min 11sec
Verify SPI Flash	2hrs 22min 10sec	2hrs 10min 28sec	23min 05sec	14min 17sec	13min 21sec
Read SPI Flash	2hrs 34min 20sec	2hrs 23min 45sec	22min 30sec	13min 54sec	12min 55sec
Erase SPI Flash	19sec	18sec	1min 51sec	1min 49sec	1min 48sec

NOTES:

[1]SPI Flash programming has been tested on N25Q00AA and MT25QL01G/MT25QU01G devices only. Contact technical support for other SPI-Flash device support needs.

[2]FlashPro6 has longer programming times for SPI Flash devices, when compared to FlashPro5. However, readback and verification times are significantly shorter. Programming time for FlashPro6 will be improved in future releases.

[3]To successfully program the device at a high TCK frequency, appropriate steps must be taken to ensure signal integrity of the JTAG signals.

8 Appendix B. RTG4 SPLL and FPLL Calibration and Workaround

Previously, SPLL (SERDES PCIe and XAUI) and FPLL (FDDR) lost lock during temperature ramp as described in CN19009 and CN19009A. To resolve this issue, a new CoreABC sequence has been developed in Libero SoC v12.4 that includes an SPLL/FPLL workaround. The new sequence requires design changes to the initialization logic (CoreABC configuration and connections) in some cases described in the following sections.

8.1 Affected Designs and Modes

This workaround applies to designs that use the following components and modes. These designs require upgrading to the latest core version. A few design changes may also be needed for designs that use components without built-in initialization. Refer to the following sections for details.

Libero Catalog Display Name	Component Name	Mode	See Page
RTG4 High Speed Serial Interface (PCIe, EPCS & XAUI)	PCIE_SERDES_IF	PCIE XAUI	52 54
High Speed Serial Interface 1 - EPCS and XAUI - with Initialization	PCIE_SERDES_IF_INIT	XAUI	54
RTG4 High Speed Serial Interface (EPCS & XAUI)	NPSS_SERDES_IF	XAUI	54
High Speed Serial Interface 2 - EPCS and XAUI - with Initialization	NPSS_SERDES_IF_INIT	XAUI	54
RTG4 DDR Memory Controller	RTG4FDDRC	DDR	55
RTG4 DDR Memory Controller with initialization	RTG4FDDRC_INIT	DDR	55

This workaround also applies to designs that use the following components that support built-in initialization. These designs require upgrading to the latest core version. Thereinafter, no user action is required because the *_INIT components handle the new CoreABC sequence and design updates automatically for SPLL/FPLL Lock polling.

Libero Catalog Display Name	Design	Mode
High Speed Serial Interface 1 - EPCS and XAUI - with Initialization	PCIE_SERDES_IF_INIT	XAUI
High Speed Serial Interface 2 - EPCS and XAUI - with Initialization	NPSS_SERDES_IF_INIT	XAUI
RTG4 DDR Memory Controller with initialization	RTG4FDDRC_INIT	DDR

8.2 The New CoreABC Sequence

The following steps describe where the new SPLL/FPLL workaround is added in the CoreABC program/sequence.

8.2.1 SPLL (for SERDES PCIe/XAUI PCS to Fabric Interface)

Refer to the *abc.txt file generated for each block for the actual CoreABC program/sequence under: project/component/work/<PCIE or NPSS SERDES component>/PCIE (or NPSS)_SERDES_IF_0

- Configure PMA, system registers.
- De-assert PHY_RESET_N.
- Perform the SPLL workaround, which includes polling for SPLL_LOCK output from SERDES.
- De-assert CORE_RESET_N.
- PCIE mode only: Configure PCIE registers.
- Assert INIT_DONE.

8.2.2 FPLL (for FDDR)

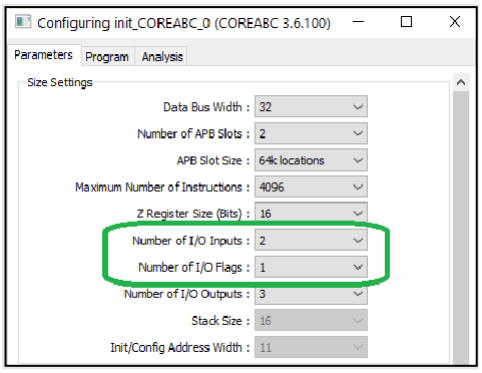
Refer to the *abc.txt file generated for each block for the actual CoreABC program/sequence under: project/component/work/<FDDR component>/FDDRC_0)

- Perform the FPLL workaround, which includes polling for the FPLL_LOCK output from FDDR.
- Configure the DDR registers.
- Wait for controller ready and memory settling time, and then assert INIT_DONE.

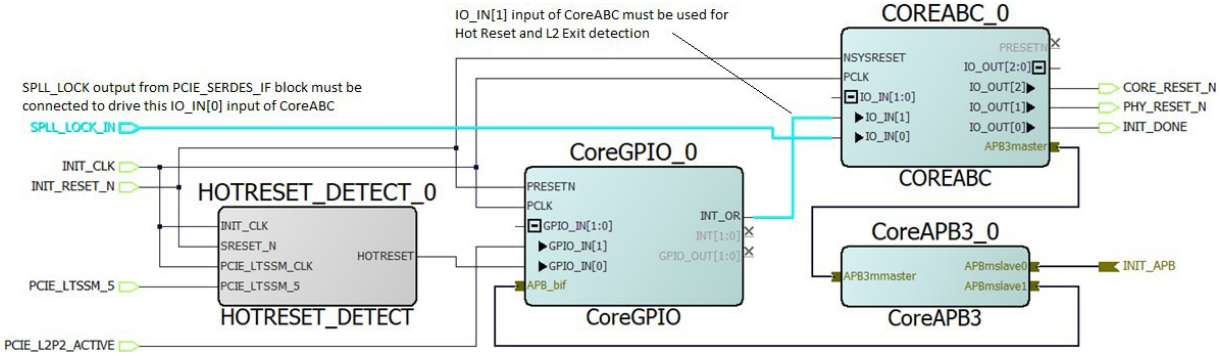
8.3 Design Updates for PCIE_SERDES_IF in PCIE Mode

After upgrading to the latest PCIE_SERDES_IF core version, copy the latest CoreABC program/sequence from the generated *abc.txt file and paste it in the **Program** tab of CoreABC that is part of the assembled initialization logic.

In addition, update the CoreABC configuration in the **Size Settings** section of the **Parameters** tab (refer to the following figures):



	Parameter	Updated Setting	Comment
	Number of I/O Inputs	2	IO_IN[0] = SPLL_LOCK IO_IN[1] = Hot Reset and L2 Exit detection logic. If your design does not use the Hot Reset and L2 Exit detection logic, tie IO_IN[1] to GND
	Number of I/O Flags	1	

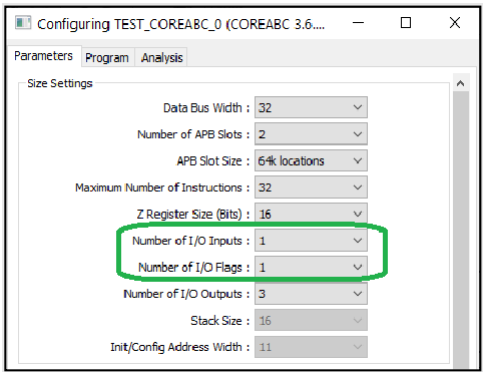


8.4 Design Updates for PCIE_SERDES_IF or NPSS_SERDES_IF in XAUI Mode

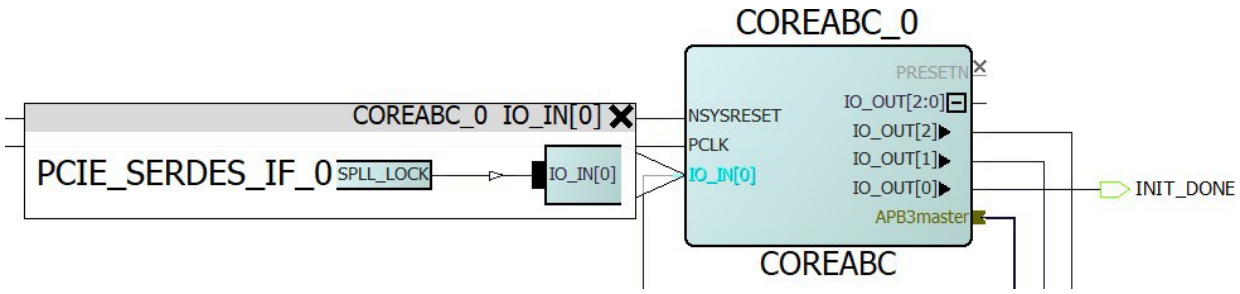
Perform the following changes only when using the SERDES blocks without built-in initialization (PCIE_SERDES_IF or NPSS_SERDES_IF). These changes are handled automatically when using the SERDES blocks with built-in initialization (PCIE_SERDES_IF_INIT or NPSS_SERDES_IF_INIT).

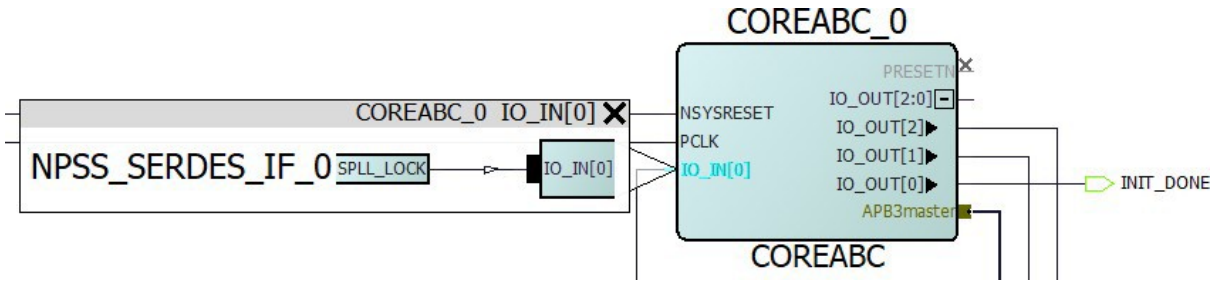
After upgrading to the latest PCIE_SERDES_IF or NPSS_SERDES_IF core version, copy the latest CoreABC program/sequence from the generated *abc.txt file and paste it in the **Program** tab of CoreABC that is part of the assembled initialization logic.

In addition, update the CoreABC configuration in the **Size Settings** section of the **Parameters** tab (refer to the following figures):



Parameter	Updated Setting	Comment
Number of I/O Inputs	1	IO_IN[0] = SPLL_LOCK
Number of I/O Flags	1	





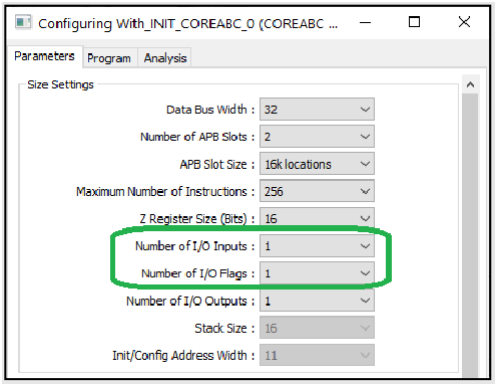
8.5 Design Updates for RTG4FDDRC

Perform the following changes only when using the DDR memory Controller without built-in initialization (RTG4FDDRC). These changes are handled automatically when using the DDR memory Controller with built-in initialization (RTG4FDDRC_INIT).

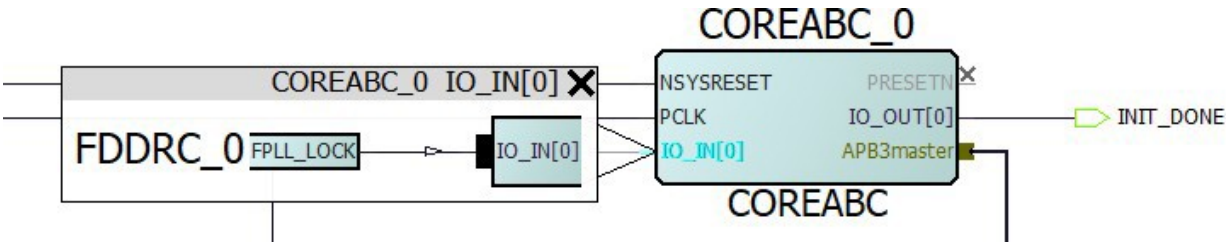
After upgrading to the latest RTG4FDDRC core version, copy the latest CoreABC program/sequence from the generated *abc.txt file and paste it in the **Program** tab of CoreABC that is a part of the assembled initialization logic. Polling for FPLL_LOCK remains an existing requirement for FDDR, so there are no new design modifications required. The new sequence has moved the polling for FPLL_LOCK earlier in the sequence, before the DDR controller register initialization starts.

Refer to the following figures for CoreABC configuration requirements in the **Size Settings** section of the

Parameters tab



Parameter	Updated Setting	Comment
Number of I/O Inputs	1	IO_IN[0] for FPLL_LOCK
Number of I/O Flags	1	



8.6 Design Updates for RTG4FCCCECALIB

Clocks from RTG4FCCCECALIB that are used to drive the CLK_BASE of SPLL (SERDES blocks) or FPLL(FDDR) follow a cascaded PLL scenario. When building the design, it is the user's responsibility to make sure that CoreABC program execution does not start until CCC_0/1_LOCK of RTG4FCCCECALIB is asserted.

9 Documents Updated in this Release

The following documents have been updated for the 12.4 release.

- Libero SoC Design Flow UG
- PolarFire FPGA Design Flow UG
- I/O Editor UG
- SmartTime static timing analyzer UG
- Timing Constraints Editor UG
- TCL Command Ref UG - SF2, IGLOO2, RTG4
- TCL Command Ref UG - PolarFire
- FlashPro Express UG
- Macro Library UG - SF2, IGLOO2
- Libero SoC Linux Environment Setup UG
- SF2 Micro RAM Configuration UG
- SF2 Two-port Large SRAM Configuration UG
- SF2 Dual-port Large SRAM Configuration UG
- RTG4 DDR Memory Controller Configuration User Guide
- RTG4 DDR Memory Controller with Initialization Configuration User Guide
- RTG4 High Speed Serial Interface(ECPS and XUAI) Configuration User Guide
- RTG4 High Speed Serial Interface (ECPS and XUAI) with Initialization Configuration User Guide
- RTG4 High Speed Serial Interface (PCIe, ECPS and XUAI) Configuration User Guide
- RTG4 Two-port Large SRAM ConfigurationUG
- RTG4 Dual-port Large SRAM Configuration UG
- SF2 MSS eNVM Configuration UG
- Libero SoC Secure IP Flow User Guide