Libero® SoC v2021.2

Release Notes

Introduction

The Libero® system-on-chip (SoC) v2021.2 unified design suite is Microchip's flagship FPGA software for designing with Microchip's latest power efficient Flash FPGAs, SoC FPGAs, and rad-tolerant (RT) FPGAs. The suite integrates industry-standard Synopsys Synplify Pro synthesis and Mentor Graphics Modelsim Pro simulation with best-in-class constraints management, debug capabilities, and secure production programming support.

Use Libero SoC v2021.2 to design with the following Microchip FPGAs:

- RTG4™
- SmartFusion®2
- IGLOO® 2
- PolarFire
- RT PolarFire
- PolarFire SoC

To design with Microchip's older Flash FPGA families, use Libero SoC v11.9 and its 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 v2021.2 does not support Classic Constraint Flow. IGLOO2, SmartFusion2, and RTG4 projects using the "Classic" flow cannot be opened in this release. For information about how to migrate Classic Constraint Flow projects to the Enhanced Constraint Flow, refer to Migrating an Existing Project Created with Classic Constraint Flow to Enhanced Constraint Flow.

Related Release Notes

In addition to these release notes, you may find the information in the following release notes helpful.

- PolarFire SoC MSS Configurator v2021.2 Release Notes
- Programming and Debug Tools v2021.2 Release Notes
- SmartHLS v2021.2 release notes

Table of Contents

1. Libero SoC Software Release Notes

These release notes contain important information about the Libero® system on chip (SoC) v2021.2 unified design suite.

1.1 Customer Notification (CN) Support

Libero SoC v2021.2 includes changes that address certain important issues.

1.1.1 CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core

With Libero SoC v2021.2, the RTG4 FCCC with Enhanced PLL Calibration core will be updated to improve robustness in various user configurations. The following changes are being implemented:

- Update the `PLL_ARST_N` input to `PLL_RST_N` and default to synchronous relationship with `CLK_50MHZ`, with a user option to revert to asynchronous assertion.
- Add `PLL_RST_N` reset release synchronization to ensure that reset release is always synchronous to `CLK_50MHZ`.
- Convert all resets internal to the PLL calibration soft IP into synchronous resets.
- Add synthesis directives to preserve/keep the PLL calibration soft IP's FSM state register and illegal state detection logic.
- Enable Single-Event Transient (SET) mitigation for all Flip-Flops (FFs) in each core instance via a Netlist Design Constraint (`.NDC`).
- When user dynamic configuration is enabled, drive the APB configuration interface with `CLK_50MHZ` input and remove option for a separate, user-supplied `APB_S_PCLK` input.
- Add a 500us timeout counter on the PLL calibration soft IPs `LOCK_WAIT FSM` state.

For more information, see CN19009C.

1.1.2 Customer Advisory Notification: SmartFusion2/IGLOO2/RTG4 SERDES PCIe AHBLite Issue

In SmartFusion2, IGLOO2, and RTG4 families, under specific PCIe traffic patterns, SERDES PCIe AHBLite master interface to fabric functions improperly, requiring design conversion to an AXI-based SERDES PCIe configuration with a new FPGA fabric-based AXI-to-AHBLite bridge soft IP.

- An issue was discovered when using the AXI-to-AHBLite master bridge inside the PCIe block of the SERDES.
- Under specific PCIe traffic conditions, the AXI-to-AHBLite master bridge embedded within the SERDES block does not load its burst counter properly, which results in early burst completions.
- Subsequently,the PCIe transaction layer packet (TLP) logic generates incorrect read completion TLPs which will eventually result in the termination of the PCIe read TLP completions. The system behavior during this scenario depends on the root port used.

Traffic conditions known to cause this issue include:

- Write transactions with open read transactions.
- Read transactions with open write transactions.

For more information, see JAON-08OHTZ048.

1.1.3 CYER-05NWLS164: PolarFire Timing Data Update

This customer notification applies to FPGA Static Timing Analysis (STA) data used with Microchip's PolarFire FPGAs. As part of continuous improvement efforts, Libero SoC v2021.2 has been updated to improve the accuracy of the PolarFire timing data in two specific cases, described in the following section.

Prior to Libero SoC v2021.2, the two PolarFire timing paths below used delay values that were slightly underestimated.

- Unused IP Interface Logic Elements used as regular combinatorial logic elements have LUT4 B input to Y output delay that was underestimated by up to 35 ps worst-case. IP Interface Logic Elements are used to connect hard IP blocks in the FPGA fabric, such as RAMs and Mathblocks, to user logic. When the associated hard IP blocks

are unused in a design, Libero SoC can re-use the IP Interface LUTs and SLEs for user fabric logic. In this scenario, the 4-input LUT input B to output Y path can be used for user logic.

- Mathblock input registers have enable pin setup time that was underestimated by up to 77 ps worst-case. Examples of Mathblock input register enable pins are `A_EN`, `B_EN`, `C_EN`, and `D_EN`.

For more information, see CYER-05NWLS164.

1.2 New Device Support

1.2.1 PolarFire

1.2.1.1 MPF050T/TS/TL/TLS

Libero SoC v2021.2 introduces MPF050T/TS/TL/TLS devices in two packages, FCSG325 and FCVG484, supported with Advance timing and power.

Table 1-1. PolarFire Advance Timing and Power

Part Number	Speed Grade/Temperature
MPF050T/TL	EXT STD 1.0V/1.05V
MPF050T/TL/TS/TLS	IND STD 1.0V/1.05V
MPF050T	EXT -1 1.0V/1.05V
MPF050T/TS	IND -1 1.0V/1.05V

1.2.1.2 MPF300TS FC784 MIL

Libero SoC v2021.2 also adds the FC784 MIL package for the MPF300TS device.

1.2.2 PolarFire SoC

1.2.2.1 MPFS025T/TS/TL/TLS Preliminary Timing and Power

Libero SoC v2021.2 extends Preliminary timing and power support for MPFS025T/TS/TL/TLS devices.

Table 1-2. PolarFire SoC Preliminary Timing and Power

Part Number	Speed Grade/Temperature
MPFS025T/TL	EXT STD 1.0V/1.05V
MPFS025T/TL/TS/TLS	IND STD 1.0V/1.05V
MPFS025T	EXT -1 1.0V/1.05V
MPFS025T/TS	IND -1 1.0V/1.05V

1.3 Software Features and Enhancements

1.3.1 Smart High-Level Synthesis (SmartHLS) general release v2021.2

Libero SoC v2021.2 includes the first general release of Smart High-Level Synthesis (SmartHLS) tool.

SmartHLS is an Eclipse-based integrated development environment that accepts C++ software code as input and generates a SmartDesign IP component (Verilog HDL) as output. Hardware engineers can instantiate the generated SmartDesign IP component in the SmartDesign canvas available in Libero SoC design suite to build an FPGA system.

The SmartHLS software requires a free license that you can request in the Microsemi SoC Portal. Use the following procedure to request the free license.

1. Log into the Microsemi Customer Portal (http://soc.microsemi.com/portal) and click the **License and Registration** link.
2. Click **Request Free License**.
3. Click **SmartHLS one-year Disk ID License for Windows** or **SmartHLS one-year Floating License for Windows or Linux Server**.
4. On the next page, enter the hard Disk ID or MAC ID of the machine based on the selected license option, and then click the **Submit** button.
   A new `license.dat` file will be emailed to you, enabling you to use SmartHLS.
5. When you receive the `SmartHLS license.dat` file, open it, copy the content, and add it to your existing Libero software license file with the similar Libero license option (Disk ID or MAC ID). For example, if you generated a node locked license for the SmartHLS tool, add the SmartHLS license file content to the Libero node locked license.

   For more details, see the:
   - Online SmartHLS v2021.2 release notes. The release notes can also be found in the Libero installation under `SmartHLS-2021.2/docs/releasenotes.html`.
   - Microchip SmartHLS User Guide.

1.3.2 SmartDesign Enhancements

Libero SoC v2021.2 adds new features to the SmartDesign tool enhancing the front-end design entry capability.

- Hierarchical SmartDesign Creation from within the Canvas
- Full Copy/Cut/Paste Support within a SmartDesign and across SmartDesigns
- New Smart Search and Connect feature replaces Quick Connect
- Synthesis Attributes support

Note: Copy and paste functions for HDL+ core instances are not yet supported and should not be used in the release. This will be fixed in Libero SoC v2021.3

For more details, see the SmartDesign User Guide.

1.3.3 Bitstream Digest

Prior to Libero SoC v2021.2, the calculated digest of each bitstream component (Fabric, Security, eNVM, etc.) that gets logged during bitstream export did not include metadata such as the software version, design version, design name, and advanced bitstream options for PolarFire SoC. Libero SoC v2021.2 calculates the digest over the entire bitstream, including the metadata. This is embedded inside the bitstream in plaintext, logged during bitstream generation and export tools in Libero SoC, and shown in the FlashPro Express UI by hovering over the info icon of the device. This feature is available for SmartFusion2, IGLOO2, PolarFire, RT PolarFire, and PolarFire SoC families.

For more details, see the Libero SoC Design Flow User Guide for the specific device family.

1.3.4 Out-of-Context Derive Constraints Utility for Custom Flows

Libero SoC v2021.2 adds a stand-alone tool to derive constraints for running synthesis outside the Libero project (custom design flow). The tool needs a Tcl script that specifies the HDL source and SDC timing constraints from the various configurators to generate SDC files for both synthesis and back-end tools.

For more details, see the Custom Flow User Guide.

1.3.5 Timing Constraint Enhancements: Multi-cycle Start and End Options

Libero SoC v2021.2 adds support for `-start` and `-end` options to the `set_multicycle` SDC timing constraint. These options allow users to specify which clock period is applied for the additional cycles.

- For `-end`, the capturing clock period is applied, which is the default for setup analysis.
- For `-start`, the launching clock is applied, which is the default for hold analysis.

Existing designs without `-start` or `-end` will not be affected, as the defaults are unchanged.

For more details, see the SmartTime Static Timing Analyzer User Guide.

1.3.6 Ability to Create and Select Synthesis and Identify Implementations

Libero SoC v2021.2 adds the ability to create and select Synthesis and Identify implementations from Libero's synthesis configuration.

Running synthesis on a newly created Synthesis implementation will make it active. After creating a new Identify implementation, invoke SynplifyPro interactively, launch Identify Instrumentor on the selected implementation, instrument debug signals, and run synthesis to activate the Identify implementation in Libero.

Selecting a previous implementation will not restore the synthesis result or its configuration options. Update the synthesis configuration and rerun synthesis.

For more details, see the Libero SoC Design Flow User Guide.

1.3.7 SynplifyPro and Identify

The SynplifyPro and Identify tools bundled in Libero SoC v2021.2 have been upgraded to version R-2021.03M. Besides enhancements and fixes, SynplifyPro is now fully supported on Ubuntu 18.04.

Generated names of signals and ports get renamed without special characters.

When the new attribute `syn_safe_cdc` is applied on a CDC path, it ensures that SynplifyPro will not add `syn_preserve`, `syn_replicate`, and `syn_allow_retiming` attributes, making these registers available for absorption during RAM and MATH block inference. SynplifyPro still reports the path as Safe and its description will be `syn_safe_cdc attribute applied`. The `syn_safe_cdc` can be applied on source, destination, or both. Example:

```
reg ar /* synthesis syn_safe_cdc = 1 */;
reg zr;
reg sync;
always @(posedge clk1)
begin
    ar <= a;
end
always @(posedge clk2)
begin
    zr <= ar;
    sync <= zr;
end
```

For more details, see the *Synopsys FPGA Synthesis SynplifyPro ME R-2021.03M User Guide*.

1.3.8 Initiator/Target Nomenclature

Libero SoC v2021.2 updates the nomenclature from Master/Slave to Initiator/Target in the SmartDesign Memory map view and reports, and PolarFire SoC standalone MSS configurator.

For more details, see the Libero SoC Design Flow User Guide.

1.3.9 Standalone Synthesis Flow

Libero SoC v2021.2 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.

1.3.10 Discontinuation of ModelSim ME

Starting with Libero v2021.2, only Modelsim ME Pro is officially bundled within Libero SoC. ModelSim ME integration has been discontinued in this Libero release. Users continue to have the flexibility of using previous and/or stand-alone versions of ModelSim ME.

1.4 New Silicon Features and Enhancements

1.4.1 PolarFire, RT PolarFire, and PolarFire SoC

1.4.1.1 Lock Bits Configuration and Enhanced Configuration Report for Safety-Critical Applications

PolarFire, RT PolarFire, and PolarFire SoC devices allow you to restrict access to the Transceiver, PCIe, Transmit PLL, PLL, DLL, Lane Controller, Crypto, TVS, Tamper, and VoltageDetect configuration registers. Libero SoC v2021.2 includes the Register Lock Bits Configuration tool to lock these configuration registers and prevent them from being overwritten by initiators that have access to these registers.

An initial Configuration Lock Bit file is created when you generate FPGA array data. The file is named `<project>/designer/<design>/<design>_init_config_lock_bits.txt`. All registers or blocks are unlocked by default. Use this file as a template to make changes. Modify it to ensure that the lock bits are set to 0 for all register bits you want locked, and then save the file with a different name.

To import the Lock Bit Configuration File into a project:

1. From the Design Flow window, click **Configure Register Lock Bits**.
2. Click the **Browse** button, go to the text file that contains the Register Lock Bit settings, and import the file into the project.

   Note: Simulation support for this feature is unavailable.

   For more details, see the AC478: PolarFire FPGAs for Safety-Critical Applications.

1.4.1.2 Power Up to Functional Timing (PUFT) Data

Libero SoC v2021.2 adds information about Power Up To Functional timing (PUFT) data in the Design Initialization Data and Memories report. To indicate the completion of initialization of each block (PCIe, XCVR, and RAM), a signal is asserted as part of device initialization after power-up. For example, the `PCIE_INIT_DONE` signal is asserted after all the PCIe-related registers are configured. The `DEVICE_INIT_DONE` signal from `PF_INIT_MONITOR` is the last signal asserted.

For more details, see the PolarFire FPGA and PolarFire SoC FPGA Device Power-Up and Resets User Guide.

1.4.1.3 LVDS 1.8V GPIO Standard (LVDS18G)

Libero SoC v2021.2 introduces a distinct I/O standard called LVDS18G to support LVDS I/Os on GPIO banks with VDDI set to 1.8V.

For more details, see the following documents:

- PolarFire FPGA and PolarFire SoC FPGA User I/O User Guide
- PDC Commands User Guide for PolarFire FPGA

1.4.1.4 User Control of Output Clock Port Pattern on IOD Generic Transmit Interface

Libero SoC v2021.2 adds an option to enable user control of the output clock port (`HS_CLK`) pattern on the `PF_IOD_GENERIC_TX` interface.

For more information, see the PolarFire FPGA and PolarFire SoC FPGA User I/O User Guide.

1.4.1.5 LPDDR3 Self-refresh Entry and Exit

Libero SoC v2021.2 adds the capability of self-refresh entry and exit to LPDDR3.

For more details, see the PolarFire FPGA and PolarFire SoC FPGA Memory Controller User Guide.

1.4.1.6 `PF_QDR DOFF_N` Port and Fast Simulation Enhancements

Libero SoC v2021.2 adds an option to **Export DOFF_N Port** and enables fast simulation with the latest version of `PF_QDR`.

For more details, see the PolarFire FPGA and PolarFire SoC FPGA Memory Controller User Guide.

1.4.1.7 SPI Flash Programming Enhancements

Libero SoC v2021.2 adds the following capabilities to SPI Flash programming.

- Support for loading STAGE3 initialization client when adding SPI Bitstream client in SPI Flash. Support for exporting STAGE3 file along with SPI bitstream file in the Export Bitstream dialog box.
- Support for loading binary image (`.bin`) SPI Flash clients.
- Programming support of Gigabit SPI Flash devices: GD25S512 and GD25Q256.
- Programming support of Micron MT25QL02G SPI Flash device.
- Programming support of Winbond W25Q128JWYIQ SPI Flash device.

For more information, see the Libero SoC Design Flow User Guide.

1.4.1.8 PolarFire XCVR-Sourced Fabric Clocks and Jitter Compensation

The PolarFire XCVR can source three different clocks into the fabric:

- `TX_CLK`
- `RX_CLK`
- `REFCLK` (`FAB_REF_CLK`)

These clocks contain high-frequency jitter that is not automatically considered 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 table shows recommended values for clock uncertainty per clock, resource, and speed-grade.

Table 1-3. Recommended Values for Clock Uncertainty

Clock Type	STD	-1
`FAB_REF_CLK` on Global	275 ps	200 ps
`FAB_REF_CLK` on Regional	N/A	N/A
`TX_CLK_G` on Global	300 ps	225 ps
`TX_CLK_R` on Regional	225 ps	150 ps
`RX_CLK_G` on Global	325 ps	250 ps
`RX_CLK_R` on Regional	250 ps	175 ps

The following example shows a clock-uncertainty constraint that uses a -1 speed grade.

```
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 } ]
```

The following example shows a clock-uncertainty constraint with STD speed grade.

```
# 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]
```

The automatic management of these constraints will be added in a future release of Libero SoC.

Note: It is also important to add the other required jitter sources as clock uncertainty into your design constraints. See the datasheet for the jitter to be added for components such as PLL, DLL, and RC Oscillator. For input pins that are direct or are inputs to the DLL (but not for a PLL), the input jitter on the clock input pin must also be added to your timing constraints.

1.4.2 PolarFire SoC

1.4.2.1 Standalone MSS Configurator Enhancements

Libero SoC v2021.2 extends the PolarFire SoC standalone MSS configurator with the following enhancements:

- An option to have both SD and eMMC I/Os pre-configured if dynamic reconfiguration is needed by the application
- Package pin information for RefClk, DDR and SGMII I/Os added to the MSS configurator report
- An option to have the eMMC and CAN clock sourced from the fabric (production devices only).

For more information, see the Standalone MSS Configurator User Guide for PolarFire SoC

1.4.3 RT PolarFire

1.4.3.1 Design-Specific IBIS Models

Libero SoC v2021.2 adds the ability to generate design-specific IBIS models for RTPF500T/TS/TL/TLS devices.

1.4.4 RTG4, SmartFusion2, and IGLOO2

1.4.4.1 Dynamic CCC Placement Diagnostic

Libero SoC v2021.2 adds a diagnostic table of recommended pin assignments to guide you out of Dynamic CCC placement failures in RTG4, SmartFusion2, and IGLOO2 designs.

For more details, see the UG0586 - RTG4 FPGA Clocking ResourcesFPGA Clocking Resources User Guide.

1.4.5 SmartFusion2 and IGLOO2

1.4.5.1 Secure Production Programming Solution Key Rotation Flow

Libero SoC v2021.2 adds the ability to securely update/rotate keys in Secure Production Programming Solution (SPPS) flows for SmartFusion2 and IGLOO2 devices.

For more details, see the Job Manager User Guide.

1.4.6 RTG4

1.4.6.1 RTG4 Input Pad Performance Improvement

Libero SoC v2021.2 includes an update to the RTG4 Input pad timing model to improve an over-conservative timing margin. Prior to Libero SoC v2021.2, false violations may occur making timing closure more difficult. With this update, such false violations will be removed. Minor performance improvement may also be seen. No action is required for completed designs that have met timing across all corners. Ongoing or new designs are recommended to upgrade to Libero SoC v2021.2.

2. Migrating Designs to Libero SoC 2021.2

2.1 Design and Core Invalidation

2.1.1 RTG4FCCCECALIB Design Invalidation

RTG4FCCCECALIB designs performed with earlier Libero SoC versions will be invalidated when Libero SoC v2021.2 is invoked.

- `Error: RTG4 CCC instances listed below must be updated to RTG4FCCCECALIB core v2.2.005 or later and regenerated. For more information about this requirement, refer to RTG4 Customer Notification CN19009C on the Microchip website.`
- `INFO: Converted design '...' to pre-Synthesize state because it contains RTG4 CCC instances that must be updated. Please check the log for more information.`
- `INFO: Converted design '...' to pre-Compile state because it contains RTG4 CCC instances that must be updated. Please check the log for more information.`

2.2 Core Enhancements and Upgrades

The following table lists the core enhancements and upgrades. For more information about updating a core version, see section 2.3. Updating a Core Version.

Table 2-1. Core Enhancements and Upgrades

Core	2021.2 Version	Status	Comments
PF_DRI	1.1.104	Production	Removed the **PCIe** tab in the PF_DRI configurator. For information about how to reconfigure dynamically the PCIe configuration regiseters, see the PolarFire PCIe User Guide.
PF_IO	2.0.104	Production	The interface was corrected when configuring input and output register modes differently.

.....continued

Core	2021.2 Version	Status	Comments
PF_IOD_GENERIC_RX	2.1.104	Production	• Fixed issue when using the option to expose fractional clock parallel data port. • Previously, IOD_GENERIC_RX with a fabric ratio of 1 generated false path constraints that did not exist in the design. These constraints have been removed. • Exposes eye_width_monitor port with RX_DDRX_B_G_A configuration.
PF_IOD_GENERIC_TX	2.0.111	Production	See section 1.4.1.4. User Control of Output Clock Port Pattern on IOD Generic Transmit Interface.
PF_IOD_TX_CCC	1.0.122	Production	Repackaged with latest core version of PF_CCC.
PF_LPDDR3	2.3.113	Production	See section 1.4.1.5. LPDDR3 Self-refresh Entry and Exit.
PF_QDR	1.8.203	Production	See section 1.4.1.6. PF_QDR DOFF_N Port and Fast Simulation Enhancements.
PF_RGMII_TO_GMII	1.3.103	Production	Repackaged with the latest core versions of PF_IOD_GENERIC_RX and PF_IOD_GENERIC_TX.
RTG4FCCCECALIB	2.2.005	Production	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
RTG4FDDRC	2.0.102	Production	Enables fractional clock frequency.
RTG4FDDRC_INIT	2.0.102	Production	Enables fractional clock frequency.
SmartFusion2/IGLOO2 FDDRC	2.0.101	Production	Enables fractional clock frequency.

2.3 Updating a Core Version

Perform the following procedure to update a core version:

1. Download the latest version of the core into your vault.
2. 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**.
3. Regenerate the core.
4. Derive the Timing Constraints again from the Constraint Manager tool to use the latest generated core timing constraints.
5. Rerun the design flow.

3. Resolved Issues

The following table lists the customer-reported defects and enhancement requests resolved in Libero SoC v2021.2 that have case numbers. Resolution of previously reported “Known Issues and Limitations” are also noted in this table.

Table 3-1. Customer-reported Defects and Enhancement Requests with Case Numbers

Case Number	Description	Resolution
00967427	For SmartFusion2 and IGLOO2 Libero designs, if the customer selected an eNVM component only in bitstream along with the sanitize all eNVM pages option, the output of Generate Bitstream, Export Bitstream, and Export FlashPro Express Job produces incorrect bitstream file where running ERASE action will program eNVM instead of sanitizing eNVM.	This issue is fixed in Libero v2022.2.
493642-2828855802	Unused encryption keys (UEK2, UEK3) must be disabled in Key Mode Policy.	In the key Mode Policy UI check boxes for unused UEK2 and UEK3 (if UEK3 is supported for the device) are disabled and "off." When open existing designs with key mode policy set to "enable" for unused UEK2/UEK3 keys, the policy resets to disabled, programming tools that generate bitstream are invalidated, and the user is informed about changes and asked to re-run tools. Newly exported TCL scripts set UEK2/UEK3 key mode policy parameters to "false" (disabled) if the keys are not used. Existing TCL scripts that enable key mode policy for unused UEK2/UEK3 keys will error-out.
493642-2822456042	Show SPI Flash client type in design initialization and data memory report.	The SPI Flash client type is shown in the report.
493642-2528026909	PF: Document post-layout macros.	The Post-Layout macros are now documented in the Macro Library Guide for PolarFire.
1-33183756	`SD_SYN_ATTRIBUTES`: Need the ability to add a Synthesis Directive to a SmartDesign Component.	See section 1.3.2. SmartDesign Enhancements
493642-2834448979, 493642-2838137518	RTG4 FDDR fractional clock frequency not supported.	• Fixed in RTG4FDDRC core version 2.0.102. • Fixed in RTG4FDDRC_INIT core version 2.0.102. • Fixed in FDDRC core version 2.0.101.

.....continued

Case Number	Description	Resolution
493642-2780872953	PF_CCC: CCC configurator showing 3-digit accuracy, but actual accuracy is 2.	Updated configurator User Interface to allow for 3-digit accuracy.
493642-2709111900	PF_QDR: Query related to DOFF signal in CoreQDR (PF_QDR).	Fixed in PF_DQR core version 1.8.203.
493642-2845898691	PolarFire IOD_GENERIC_RX "expose fractional clock parallel data" port not working.	Fixed in PF_IOD_GENERIC_RX core version 2.1.104. HS_IO_CLK port connection issue has been resolved.
493642-2817010799, 493642-2838360828	PF_IOD_GENERIC_TX: Add option to enable user control of clock pattern on HS_CLK.	Fixed in PF_IOD_GENERIC_TX core version 2.0.111.
493642-2836680172	PF_IOD_GENERIC_RX: IOD_GENERIC_RX with fabric ratio of one generates the following constraints, which do not exist in the design.	Fixed in PF_IOD_GENERIC_RX core version 2.1.104.
493642-2813090173	RTG4_FCCCECALIB: Use CLK_50MHZ for APB_S_PCLK when Dyn Config is enabled.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
493642-2839029728	RTG4_CCC: PLL with differential feedback.	Updated configurator User Interface to allow for dedicated pads to be configured in differential mode when using external feedback.
493642-2813090173	RTG4_FCCCECALIB: RTG4 Enhanced PLL IP RTG4CCCECALIB SET mitigation.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
493642-2811312932	SDC processing is taking too much time for customer's design.	
493642-2654458589	Support for -start and -end switches in set_multicycle_path constraint.	See section 1.3.5. Timing Constraint Enhancements: Multi-cycle Start and End Options.
493642-2761382645	SmartTime **Apply filter** is not working.	Apply filter activated in addition to stored filters.
493642-2859123557	Different SmartTime behavior in Libero v12.3 and v12.6 for input delay constraint.	The path reported in the user set gives higher priority to fully constrained paths.
493642-2761427558	Libero generated fdc file generates error during synthesis.	When a clock name (without get_clocks) contains a '/' don't convert the '/' to '.' into the fdc file for synthesis.
493642-2743715317	Inconsistent behavior of SmartTime in Libero v12.3 and Libero v12.4.	SmartTime stops propagating clocks when a generated clock is reached, except when a generated clock is behind another generated clock with the clock source and the master pin.

.....continued

Case Number	Description	Resolution
493642-2805318705	Differences between Design bitstream file type options while exporting FlashPro Express Job file and their impact on sNVM access.	The online help file for 'Include Plaintext Pass Keys' was updated to mention the differences between different design bitstream file types.
493642-2864361216	Add instructions to install "lsb" library for running license daemons on Linux in addition to libc6-i386 (Ubuntu).	Added an instruction under Libero SoC Installation Instructions in Ubuntu for libraries to install.
493642-2706229069	Correct the Tcl command report for timing in the user guide.	The Tcl Commands Reference Guide has been corrected for reporting inter-clock domain timing paths.
493642-2822285667	Improve Libero message during DRC check.	Updated the Self-Instantiation description under Design Rules Check in online help.
493642-2816191143	Error message correction needed.	Updated the error code message for error code `0x802B` in the FlashPro Express User Guide.
493642-2720279874, 493642-2724143412, 493642-2731049050, 493642-2729244859, 493642-2728846999, 493642-2730631859	Synthesis fails in secure IP flow for Libero v12.2 and Libero v12.3.	Removed the unwanted commands in Libero SoC Secure IP User Guide.
493642-2835376443	SET filter for `GRESET` is supported on dedicated I/O and fabric-routed signal.	The information about the GRESET macro was added to the RTG4 Macro Library User Guide.
493642-2829401262	Chip Planner crashes after renaming the regions twice.	The crash has been resolved.
493642-2805280201	Libero v12.5 SP1 I/O Editor: Pressing Enter key does not work.	This issue has been resolved; pressing the return key goes to the next cell.
493642-2608388565	`SD_CONNECT`: `QUICKCONNECT`: Connecting modules using net names.	See section 1.3.2. SmartDesign Enhancements.
493642-2863195365	Exported SPI flash `.bin` file is unusable when programming SPI without Flash Pro Express.	This fixes an issue introduced in Libero SoC v2021.1 where the exported SPI Flash `*.bin` file was not useable by third-party solutions. Libero SoC v2021.2 exports SPI Flash images in raw format to allow for third-party use.
493642-2858675628	SmartPower crashes in designs with RX only transceivers.	The fix avoids the crash when the transceiver is configured in RX-only mode.
493642-2779201786	RTG4: Placement failure MSG for Enhanced CCC GLx connection needs improvement.	See section 1.4.4.1. Dynamic CCC Placement Diagnostic.

.....continued

Case Number	Description	Resolution
493642-2810172350	Memory required to run Driver Replication exceeds 16 GB.	Reduced memory consumption of Driver replication below 16 GB.
493642-2751002554	RTG4_FCCCECALIB: Synchronize Release of PLL_ARST_N to CLK_50MHZ.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
493642-2751002554	RTG4_FCCCECALIB: Add timeout in LOCK_WAIT state.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
493642-2694971261	Device Info log displays wrong CheckSum and Design name in SPI Slave programming.	Issue resolved by inspecting both TX and RX busy bits of the SPI buffers.
493642-2812502249	FlashPro6 support of W25Q128JWYIQ SPI memory device.	See section 1.4.1.7. SPI Flash Programming Enhancements.
493642-2695952363, 493642-2698474573	Support for Gigadevice flash memory.	See section 1.4.1.7. SPI Flash Programming Enhancements.
493642-2858675628	SmartPower crashes with PF RX-only XCVR.	The fix avoids the crash when the transceiver is configured in RX only mode.
493642-2848653939	SD_BIF: AXI4STREAM: Generated VHDL from SmartDesign is illegal if an AXI-Stream port is marked unused.	Updated the AXI4 Stream Bus Interface definition to handle unused signals properly.
493642-2829905835	PF_CCC: PF_IOD_TX_CCC Simulation Issue: unstable PLL_LOCK with cascaded CCCs.	PolarFire CCC simulation model updated to consider tolerable deviations at input reference clock of PF_IOD_TX_CCC.
493642-2825894420	PF_CCC: PolarFire PLL external feedback differences between Libero SoC v12.4 and Libero SoC v12.5.	PolarFire CCC simulation model updated to return the input and output clock alignments as they behaved in Libero v12.4.
493642-2843193216	PF_TVS: Simulation behavior.	PolarFire TVS simulation model updated to clear TEMP_HIGH and TEMP_LOW alarm flags by asserting TEMP_HIGH_CLR and TEMP_LOW_CLR inputs, respectively.
493642-2844075913	PF_XCVR: 12.5G XCVR simulations are not working for PolarFire.	PolarFire XCVR simulation model updated to recognize change in FBDIV and relocks TXPLL.
493642-2761489068	PF_XCVR: Simulation not working for SERDES PMA lanes through DRI.	PolarFire XCVR simulation model updated to access all XCVR lanes through DRI.
493642-2771746791	G4_XCVR: TX_CLK_STABLE is not de-asserted in simulation when REF CLK is not available.	IGLOO2 XCVR simulation model updated to de-assert the TX_CLK_STABLE signal when ref clk input is stuck at one or zero other than regular clock behavior.

.....continued

Case Number	Description	Resolution
493642-2798868524	PF_XCVR: Transceiver RX Ready is asserted and RX clock is toggling even when PCS_ARST_N is asserted in simulation.	PolarFire XCVR simulation model updated to de-assert RX_Ready and to stop RX_CLK_R clock when PCS_ARST_N is asserted.
493642-2796352746	PF_XCVR: Simulation not working when **Rx CDR Reference clock source** is set to fabric in RX-only transceiver mode.	PolarFire XCVR simulation updated to work when the Rx CDR Reference clock source is set to fabric.
493642-2845474275	PF_CCC: PolarFire CCC simulation issue.	PolarFire CCC simulation model updated to remove the misalignment between 0 and 180 phase shifted outputs.
493642-2739447777, 493642-2842563037	PF_CCC: Incorrect simulations results on PolarFire CCC.	PolarFire CCC simulation model updated to generated accurate frequencies for certain configurations.
493642-2829954025	RTG4_LSRAM: Dual Port LSRAM sim model warnings are incorrect or missing.	RTG4 LSRAM simulation model updated to print warning messages in case write collisions occurring on the same clock.
493642-2776910289	PF_DRI: PolarFire reconfiguration controller.	PolarFire DRI simulation model updated to consider PSTRB input.
493642-2816191143	Error message correction needed.	Libero/FlashPro Express now displays the correct message for signal integrity issues.
493642-2780147696	Device info log should display Step Mark, firmware version.	DEVICE_INFO now shows Die Revision and Die Firmware Version, as requested.
493642-2663319510	Incorrect BSR description on SC_SPI_SDI / SDO of SmartFusion2 and IGLOO2 BSDL file.	SmartFusion2 and IGLOO2 BSDL files now have correct descriptions for SC_SPI_SDI and SC_SPI_SDO.
493642-2855024392	PolarFire: Print TVS_MONITOR status appropriately.	DEVICE_INFO now prints TVS status for each of the four channels only if its valid.
493642-2839083505	Chain stp support.	Added support to export chain stapl that programs single Microsemi device in a JTAG chain.
493642-2777795242	Differences in the power summary between the SmartPower and Power Estimator.	Changed IO static power model in Estimator to match the Smart Power IO power model
493642-2825236133	Infer Gated Clocks from the Enabled-Registers option produce an incorrect netlist.	With the **Gated Clock** option set to ON, UDRCLK of UJTAG macro drives only one CLKINT in the instrumented design
493642-2791323207	Synthesis error occurs when the keyword **when** is used as a keyword VHDL.	Synthesis passes when the keyword **when** is used in the condition to compare two signals or variables.

.....continued

Case Number	Description	Resolution
493642-2823251067	Synplify crash on attached project (Libero v12.6).	SynplifyPro gives proper error message in the synthesis log file for the incomplete module header parameter list.
493642-2854862880	Synthesis issue with customer design.	Crash has been resolved. Synthesis passes when all SystemVerilog files are ordered correctly through Libero.
493642-2749061712, 493642-2790763501, 493642-2793376084	Compile fails.	Libero → Compile passes for the generated synthesis netlist. The new synthesis netlist has been updated to handle escape characters for the ports, instance and the net names.
493642-2759082638	Libero archive project.	HDL Core definitions are now archived properly when selecting **Project Files Only** mode.
493642-2836226625	Libero deletes the non-existent BIST parameters when TCL file is exported for SystemVerilog modules.	Fixed this issue to make sure that an incorrect list of params does not come from packages that are not related to the core. The exported TCL is now correct,
493642-2827333700	Linked File issues with Libero v12.6.	Fixed the create_link function that was incorrectly detecting the link file as local file by matching the parent project path and is importing file locally.
493642-2782286269	`PF_DDR`: Implement error message if required `IOSTD` for the DDR interface is not set.	Fixed the displaying of multi-line error text in the tool tip.
493642-2841223969	PolarFire, PolarFire SoC, RT-PolarFire, RT-PolarFire SoC: Add support for LVDS18 input and output.	See section 1.4.1.3. LVDS 1.8V GPIO Standard (LVDS18G).
493642-2856101322	IGLOO2/RTG4_SERDES: Lock user to enter all `TX_AMP_RATIO` more than 0x80.	Added proper DRC to the configurator.
493642-2613190339	SmartPower issues false warnings.	Warnings are removed because they were coming from an fabric element that has an IIP interface that connects to a clock that is not used.
493642-2831101105	Enhancement: Alter the labels in the Manage Clock Domains dialog box.	The labels **Available clock domains** and **Show the clock domains** are altered in the Clock Domain dialog box.
493642-2851337659	Hold time violation with custom volt/temp setting.	See section 1.4.6.1. RTG4 Input Pad Performance Improvement.

Table 3-2. Customer-reported Defects and Enhancement Requests (No Case Numbers)

Description	Resolution
PolarFire: SPI: add encryption status in the CDIDAM.	A new **User Security** column has been added to the SPI Flash clients table.
For PolarFire and PolarFire SoC, older core versions of PF_DRI core had a **PCIe** tab indicating that DRI could access PCIe registers. However, DRI is needed to access only the related XCVR lane used with a PCIe controller.	The latest version of PF_DRI v1.1.104 does not have the options to enable the PCIe DRI interface. Moreover, the Tcl parameters to enable the same have been removed from the core. Migration succeeds, but the old Tcl scripts must be updated to delete the corresponding Tcl parameters from the core configuration.
SD_SYN_ATTRIBUTES: Customer wants support for Synthesis Attributes added to Canvas.	See section 1.3.2. SmartDesign Enhancements.
Add PolarFire Libero support for safety-critical applications.	See section 1.4.1.1. Lock Bits Configuration and Enhanced Configuration Report for Safety-Critical Applications.
PolarFire: PF_DRI for PCIe.	Removed the **PCIe** tab in DRI configurator.
G4_PCIE SERDES_IF: Add warning when the PCIe AHBLite interface is selected.	See section 1.1.2. Customer Advisory Notification: SmartFusion2/IGLOO2/RTG4 SERDES PCIe AHBLite Issue.
PF_CCC input frequency needs more significant figures.	Updated PF_CCC configurator User Interface to allow for 3-digit accuracy.
PF_LPDDR3: LPDDR3 needs to support issuing of self-refresh command.	Fixed in PF_LPDDR3 core version 2.3.113. See section 1.4.1.5. LPDDR3 Self-refresh Entry and Exit.
PF_PCIE: PolarFire support for 64-bit non-prefetchable BAR in PCIe EP.	Updated PF_PCIE configurator to add 64 bit non-prefetchable BAR option.
RTG4_FCCCECALIB: Change to PLL_RST_N input and configurable SYNC or ASYNC reset.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
RTG4_FCCCECALIB: Supply NDC to turn on SET mitigation for each core instance.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
RTG4_CCC: Increase significant figures allowed in CCC configuration GUI.	Frequencies can now be entered with 4-digit precision.
Cascaded CCC Clock Generation is 0 ns.	Fixed the clock generation when two generated clocks are cascaded and one of them is combinational.
RTG4_FCCCECALIB: Supply NDC to turn on SET mitigation for each core instance.	See section 1.1.1. CN19009C: Improvements to RTG4 FCCC with Enhanced PLL Calibration Core.
Missing path from bi-directional port with generated clock.	When searching for paths from a generated clock set on a bidirectional port, SmartTime looks for paths going out of the FPGA (as if the port was output only) as well as the paths coming into the FPGA.
FlashPro Express v12.3: run_selected_actions TCL command not generating return code.	The description of the TCL command run_selected_actions was updated to specify that the selected action does not return an exit code.
PTOLEMI EXCEPTION: Timing cell does not exist for BANKEN_SE_ICBW.	The timing model for the cell was missing from the timing database.
FlashPro 6: Support Micron MT25QL02G SPI-Flash.	See section 1.4.1.7. SPI Flash Programming Enhancements.

.....continued

Description	Resolution
Add a refresh button to GUI to rescan and reattach to FlashPro hardware.	Enhancement has been added.
SmartDebug RTG4 SERDES reports `RX_PLL` and `TX_PLL` unlocked on Dev Kit.	The appropriate `Rx_PLL` and `TX_PLL` values are now shown in the SmartDebug tool.
`PF_XCVR`: Inconsistency between hardware and Sim behaviors with RX PLL lock.	PolarFire XCVR simulation model updated to have correct behavior on `RX_IDLE`, `RX_READY`, and `RX_VAL` in simulations in absence of serial traffic.
`PF_CCC`: Error (suppressible): (vsim-3601) Iteration limit 5000 reached at time <time> ps.	PolarFire CCC simulation model updated to stabilize internal signals before their use to avoid infinite iteration limits.
Enable SPI-Flash Programming using customer STAPL file attached to IGLOO2 device.	Fixed FlashPro Express to allow loading custom stp without device density/package, enabling customers to embed custom JTAG commands that will work on any device.
Onehot encoding flagged on a Synopsys provided VHDL library CD231: std1164.vhd"|Using onehot encoding for type mvl9plus.	If SynplifyPro encounters a `type` statement as below to declare a set of values, it gives a note even if the state machine is not synthesized. Type `MVL9plus is ('U', 'X', '0', '1', 'Z', 'W', 'L', 'H', '-', error)`
Add date and time into `.prjx` to indicate when it was last used, saved, or modified.	Starting with Libero SoC v2021.2, a `libero_setup_info.txt` file is created in the Libero project directory and updated when the `.prjx` file is modified. For details, see the Libero SoC Design Flow User Guide.
`HDL_LANGUAGE`: HDLPLUS: `LINK_FILE`: HDLPlus definition on linked files is lost when design created on Windows is opened on Linux.	Libero SoC v2021.2 resolves this issue where HDL core using HDL links are corrupted when moving the project between Linux and Windows operating systems.
`HDL_LANGUAGE`: HDLPLUS: `LINK_FILE`: SmartDesign core definition gets corrupted with link loss.	Libero SoC v2021.2 resolves this issue where HDL core using HDL links are corrupted when moving the project between Linux and Windows operating systems.
SmartFusion2: MSSUBIT settings will not be correct if any LUT is placed at location (0,0).	Corrected reporting when LUT is placed at (0,0).
`PRPF_016`: Failed to automatically place instance 'PROC_SUBSYSTEM/ pf_JTAG_Debug_0/pf_JTAG_Debug_0/ genblk1.genblk1.genblk1.UJTAG_inst'.	The merging of multiple UJTAG macros failed when the **Infer Gated Clocks from Enabled-Registers** Synthesis option was used for the design. This issue has been resolved in Libero SoC v2021.2.

4. Known Issues and Limitations

The following table lists known issues and limitations associated with Libero SoC v2021.2.

Table 4-1. Known Issues and Limitations Associated with Libero SoC v2021.2

Family	Description
	Libero
All families	`HDL_LANGUAGE: GLOBAL_INCLUDE_PATH:` When a module is present in an include file, the file is shown as a linked file inside the project. When there is a broken Global Include Path for such projects, the include file is also shown with a broken link. If the project is closed and re-created, the correct Global Include Path displays correctly. If the Global Include path under Project Settings is changed, two links are shown for include file inside the project: • One with the new Global Include Path, as shown inside the project. • The other is the broken link with the previous Global Include Path. There are no functionality issues, but an extra broken link appears in the project. Remove this link when the Global Include Path in Project Settings is changed.
All families	In SmartDesign, the copy and paste functions for HDL+ core instances within the same or different SmartDesign may not work properly with respect to parameter configuration. Workaround: Instead of using copy and paste functions for HDL+ core instances, instantiate new instances of HDL+ core in your SmartDesign from the Design Hierarchy directly and configure them as needed.

All families	In Libero flow, when multiple identify implementations are created and run initially, the error message only appears in the Synthesis log. It does not propagate to the Libero error log. The error message should appear as follows in the Libero window:
	`No instrumentation file found for Identify implementation.`
	`Please open SynplifyPro interactively to launch Identify Instrumentor`

All families	When Identify implementation is created through **Libero -> Synthesis -> Configure Options**, before running any Synthesis or creating any new implementation, open SynplifyPro interactively, instrument the design, and run Synthesis. Then return to Libero to create any new Synthesis/Identify implementation.
All families	In **Libero -> Synthesis -> Configure Options -> Active Implementation** drop-down list, if any existing Synthesis/Identify implementation is not visible, open SynplifyPro interactively and re-run the intended implementation. Upon re-run, the active implementation will appear in the **Libero -> Synthesis -> Configure Options dialog -> Active Implementation** drop-down list.
All families	If multiple implementations are created in Libero/Synplify flow, each time the active implementation is switched, Synthesis must be rerun for Libero to fetch the intended .vm netlist (for active implementation).

.....continued

Family	Description

PolarFire	Since the Synplify 2020.09MSP1 release (Libero v20201.1), an enhancement to Synthesis Compiler is related to the initial value support for memory initialization and is not technology specific. Testcases with MiV core fail in Synthesis with following error message: `@E: CS162 :"./component/work/MIV_RV32IMA_L1_AXI_C0/MIV_RV32IMA_L1_AXI_C0_0/core/miv_rv32ima_l1_axi_data_arrays_0_ext.v":81:24:81:38|Loop iteration limit 2000 exceeded - add '//synthesis loop_limit 4000' before the loop construct`
	Workaround: In the **Libero GUI -> Synthesis -> Configure Option -> Additional Parameters** box, add:
	`set_option -looplimit 4000`
	OR
	Add the following `SYNPLIFY_OPTIONS` options:
	`configure_tool -name {SYNTHESIZE} -params {BLOCK_MODE:false} -params {BLOCK_PLACEMENT_CONFLICTS:ERROR} -params {BLOCK_ROUTING_CONFLICTS:LOCK} -params {CLOCK_ASYNC:800} -params {CLOCK_DATA:5000} -params {CLOCK_GLOBAL:2} -params {PA4_GB_COUNT:24} -params {PA4_GB_MAX_RCLKINT_INSERTION:16} -params {PA4_GB_MIN_GB_FANOUT_TO_USE_RCLKINT:1000} -params {RAM_OPTIMIZED_FOR_POWER:0} -params {RETIMING:false} -params {SEQSHIFT_TO_URAM:0} -params {SYNPLIFY_OPTIONS: set_option -looplimit 4000} -params {SYNPLIFY_TCL_FILE:}`
	Set this option and run Synthesis. It should pass.

PolarFire	Incorrect PUFT timing is reported when SPI Flash is using as Initialization client.
PolarFire and PolarFire SoC	If the fabric is programmed, but there are I/Os that are not used in the user's design, the unused I/O pins will glitch at power-up when the Libero SoC-generated defaults are used.
PolarFire, PolarFire SoC	`PF_DRI` had a **PCIe** tab indicating that DRI could access PCIe registers. However, DRI is needed to access only the related XCVR lane used with a PCIe controller. The latest version of `PF_DRI` v1.1.104 does not have the options to enable the PCIe DRI interface. Moreover, the Tcl parameters to enable the same have been removed from the core. Migration succeeds, but the old Tcl scripts must be updated to delete the corresponding Tcl parameters from the core configuration.
PolarFire and PolarFire SoC	In Libero designs containing SPI bitstream clients (IAP, Auto Update, or Recovery/Golden) along with a STAGE3 initialization client in the **SPI Flash** tab of the Configure Design Initialization Data and Memories tool, the output of the Export SPI Flash Image tool does not contain data for the STAGE3 initialization clients. **Workaround**: If possible, use the Generate SPI Flash Image and program in Libero or Export FlashPro Express job with SPI Flash and program via FlashPro Express.

.....continued

Family	Description
RTG4	Libero hangs or crashes on IP component generation for RTG4 FCC with Enhanced PLL calibration for some specific configurations. This issue has been seen since the Libero SoC v12.5 release.
RTG4	This issue affects designs where the **Output Re-sync after Lock** option is configured to **Held Output in Reset (output low) after power-up. Released and resynchronized with the PLL reference clock after the PLL locked**. If the `CCC_#_GL#` in BYPASS MODE is used to source the 50 MHz `ECALIB` clock, a deadlock occurs and there is no toggling of the output clocks or having the `LOCK` signal go high. A cyclic dependency is observed between `ECALIB` and the PLL IPs. The PLL is configured for outputs held in reset after power-up and is not released until the PLL locks, without sending output clock to `ECALIB`. This results in `ECALIB` not generating the proper lock enable signal that the PLL requires to generate the output signals.
RTG4	When the **Held output in reset (output low) after power-up. Released and resynchronized with the PLL reference clock after the PLL locked** option is set, clock appears at the output before the `LOCK` signal is asserted. Possible Solution: The synchronizer used for `LOCK` signal should not use the `GLx` output clock. A free-running clock can be used (for example, `CLK_50MHz`). The output of the three-stage synchronizer should be used to drive both the `LOCK` signal and the `GLx_Yx_EN` signal.
SmartFusion2, IGLOO2	In Libero, even if the keys in key Mode Policy are disabled and key Mode policy is used, the disabled keys can continue to be used to ERASE and VERIFY bitstreams. **Workaround:** Disable keys in Security Policy Manager instead of trying to disable them in key Mode Policy.

SmartFusion2	For SmartFusion2 devices, when the firmware is exported from Libero SoC v2021.1, the tool generates a `Top_hw_platform.h` file. Importing this file into SoftConsole may result in several errors similar to the following:
	`.....`
	`In file included from ../main.c:13:`
	`D:\Del\HW_Platform\Test_Hw\drivers\CoreUARTapb/`
	`core_uart_apb.h:175:1: note:`
	`declared here`
	`175 | UART_init`
	`| ^~~~~~~~~`
	`make: *** [subdir.mk:20: main.o] Error 1`
	`"make all" terminated with exit code 2.`
	`Build might be incomplete.`
	`.....`
	This is a new bug with the exported `Top_hw_platform.h` file in the current Libero release. Customers must fix this error manually by replacing `#define TOP_SB_0/COREUARTAPB_0_0 0x5000_0000U` with `#define COREUARTAPB_0_0 0x50000000U`.

.....continued

Family	Description
RTG4	I/O Editor display resistor pull for P and N pin when LVDS failsafe (Dynamic ODT) is enabled. For RTG4, the resistor pull information for P and N pin are both shown as pull up or pull down. They should be shown as pull up for one pin and pull down for the other pin.
	Synthesis/Simulation
PolarFire	`CoreQDR_PF`: Low data rate 500 Mbps/250 MHz fails in QDR II + Xtreme Device in simulation, but passes on the board.
PolarFire	When automatic compile point is enabled, Synthesis passes successfully, but Place and route errors our pointing to the derived constraint file: `Error: SDC0025: C:\validate\PF_Mi_V_Tut\constraint\PROC_SUBSYSTEM_derived_constraints.sdc:17: Invalid false path constraint: the -through value [get_nets {AXI4_Interconnect_0/ARESETN* }] is incorrect.` Workaround: Either turn off (uncheck) **Automatic Compile Point** while running Synthesis through the **Libero > Synthesis > Configure Options** dialog box. Keep the remaining options and all constraints as they are. OR Keep everything the same, but update the line below in the file `*derived_constraints.sdc` to use `get_pins` instead of `get_nets`. `set_false_path -through [ get_pins { AXI4_Interconnect_0/ARESETN* } ]`
All families	This issue was seen with the SynplifyPro R2021.03M release. On Ubuntu platform, when Synplfiy Pro is invoked, it lists packages and libraries on the terminal. This message can be safely ignored. Similarly, choosing **Libero -> Edit Profile -> Synthesis** displays a list of packages and libraries on the Ubuntu platform. Click **OK** and everything will work fine.
	Timing/Power
PolarFire	The max clock frequency on the regional clocks (`RX_CLK_R/TX_CLK_R` on the serdes and `RX_CLK` on `lanectrl`) in `MPF300XT` do not match the datasheet.
PolarFire	The violation report shows violations (red cross), but the timing report does not (green check). This indicates there is indeed a timing violation and the red cross is correct.
SmartFusion2	On Linux, the timing tool crashes when trying to verify that paths exist between the specified `-from` and `-to` when using `[ all_registers -clock some_clock ]`. **Workaround:**Use `[ get_clocks some_clock ]` instead of `all_register -clock some_clock`.
	SmartDebug
PolarFire, PolarFire SoC	In SmartDebug, when Dual mode PCIe design is considered, Data Rate, Link Width, and other values are shown only for PCIe1, but not for PCIe0.

.....continued

Family	Description
PolarFire, PolarFire SoC	When a dual-mode PCIe design is considered in SmartDebug, the following issues are observed in the PCIe debug feature: • For dual-PCIe designs with `PCI0` and `PCIe1` controllers, only `PCIe1` appears in the UI. `PCIe0` is not shown. • When **PCIe0 Lane** is selected, only the `LTSSM` state is shown for `PCIe1`. The `LTSSM` state for `PCIe0` is not shown. • Data Rate, Link Width, and other settings are shown only for `PCIe1`, but not for `PCIe0`.

PolarFire	Design using DDR controller with FHB enabled may crash during the compiling stage with the following error:
	`Internal Error:`
	`Assertion Failed in:`
	`"F:/release/sn/capture_sjsrvts05_mica_jen_sn/nsrc/hld/lib/gdevbase/src/gdev/gdevchip.cxx", line 1598.`
	`Please call technical support for assistance, providing the above diagnostic information.`

Programming	
All families	When performing any action with FlashPro 6, the following error might occur: `Error: Linux FP6 programmer - cyusb_bulk_transfer error.` This error could be caused by the programmer being out of sync with the software application. **Workaround:** Unplug the USB cable from either the programmer or the host PC, and then reconnect it to reset the programmer.
All families	When performing any action with the iCicle kit, the following error might occur: `Error: Failed to open eFP6 HID handle.` This error could be caused by the programmer being out of sync with the software application. **Workaround:** Unplug the USB cable from either the programmer or the host PC, and then reconnect it to reset the programmer.
PolarFire SoC	`eFP6`: Some testcases fail with the following error message: `Error: Failed to send data to eFP6. Err = -4|Error: eFP6 connection failed.` **Workaround:** Power cycle the iCicle board and rescan the programmers.
PolarFire SoC	For PolarFire SoC Libero designs that contain eNVM, running `VERIFY_DIGEST`, the device will fail after programming with the error message `eNVM digest verification: FAIL`. **Workaround:** Deselect the procedure **DO_ENABLE_ENVM** in the `VERIFY_DIGEST` action.

.....continued

Family	Description
PolarFire SoC	For PolarFire SoC Libero designs that generate or export eNVM-only bitstreams, the generated bitstream file/job will include an `ERASE` action that is not applicable and does nothing. Affected releases are Libero v12.4 and later. For Libero v12.6, this issue applies for eNVM only cases with no eNVM sanitization option configured.
PolarFire SoC	For Libero designs with sNVM clients configured with no custom user security options selected and program this design on device, modifying an sNVM client content and sNVM client Fabric/MSS read/write permissions and running `VERIFY` action fails with the error message `Failed to verify Security` instead of `Failed to verify SNVM`. Affected releases are Libero v12.4 and later.

SmartFusion2, IGLOO2, RTG4, PolarFire, PolarFire SoC	Some users see the following error message during programming:
	`Error: programmer 'S201QVCGH' : device 'RT4G150' : FP5 Scan: JTAG_ExecuteCommandSequence`
	`FP5: Error code = 4 - General device IO error.`
	`......`
	`Error: programmer 'S201QVCGH' : device 'RT4G150' : Executing action VERIFY FAILED.`
	`Error: programmer 'S201QVCGH' : FP5 SyncWithProgrammerAtPort: OpenSpecifiedHiSpeedDeviceBySerialNumber - PortB FP5: Error code = 2 - Device not found.`
	`......`
	This is most likely a USB connection issue. If for some reason the connection is interrupted, this error occurs.
	The error message may be different, depending on where the packet is dropped during verify. However, the error code will always be set to 4, which is “General device IO error.”

SmartFusion2, IGLOO2	Export Bitstream with SVF checked may fail in the Libero v2021.2 release. The actions `do_read_prog_info()` and `dump_programming_interface()` are not defined for SVF in the programming algorithm, but are always referenced in the `PROGRAM` action. Do not reference either function in the `PROGRAM` action when the file exported is SVF.
Installation and System Limitations	
All families	The following link provides information about FlexNet error codes: https://knowledge.autodesk.com/search-result/caas/sfdcarticles/sfdcarticles/Common-FlexNet-error-codes.html
Linux Package Required	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`

.....continued

Family	Description
All families	Many antivirus and Host-based Intrusion Prevention System (HIPS) tools flag executables and prevent them from running. To avoid this block, modify your security setting by adding exceptions for specific executables in the antivirus tool. For assistance, contact the tool provider. 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, Libero SoC, ModelSim ME, and/or Synplify Pro ME operation may or may not be affected. To ensure that all public releases of Libero software are not infected, each software package is tested with several antivirus applications before being released. To ensure further security, the Microchip software development and testing environment is protected by antivirus tools and other security measures.

5. System Requirements

The Libero SoC v2021.2 release has the following system requirements.

5.1 Supported 64-bit Operating Systems

- Windows 10 OS
- Red Hat Enterprise Linux 7.2-7.6 and Red Hat Enterprise Linux 8.x
- CentOS 7.2-7.6
- OpenSUSE Leap 42.3 (SLES 12.3 equivalent)
- Ubuntu 18.04 (Identify and Modelsim ME Pro do not directly support the Ubuntu platform. FlashPro5 programmer is not supported with Ubuntu. The Job Manager tool is not supported.)

Note: Setup instructions for using Libero SoC v2021.2 on Red Hat Enterprise Linux OS, CentOS, or Ubuntu are available in the *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.

Support for the following operating systems have ceased. For more information, see PDN20028 and PDN21004.

- Red Hat Enterprise Linux 6.6-6.11
- CentOS 6.6-6.11
- Windows 7

5.2 Random-Access Memory (RAM) Requirements

Minimum of 16 GB RAM.

6. Download Libero SoC Software

The following are available for download:

- Libero SoC v2021.2 (Linux)
- Libero SoC v2021.2 (Windows)
- Libero licenses
- MegaVault (Linux)
- MegaVault (Windows)
- Program & Debug (Linux)
- Program & Debug (Windows)
- Stand-alone MSS configurator installer (Linux)
- Stand-alone MSS configurator installer (Windows)
- snpslmd (Linux)
- snpslmd (Windows)

Note: Windows installations require administrative privileges.

After a successful installation, clicking **Help-> About Libero** shows Release: 2021.2.

7. Documents Updated in This Release

The following documents have been updated for the v2021.2 release:

Document Title	Description
Libero SoC Design Flow User Guide	Explains how to use the Libero SoC Project Manager with PolarFire, SmartFusion2, IGLOO2, and RTG4 devices.
Chip Planner User Guide	Describes how to use Chip Planner for design floorplanning.
I/O Editor User Guide	Explains how to use the I/O Editor to edit constraints.
Netlist Viewer User Guide	Describes Netlist Viewer invoked stand-alone from the Libero Design Flow window and available in pre-synthesis, post-synthesis, and post-layout phase of design flow.
Netlist Viewer Interface User Guide	Describes the Netlist Viewer Graphical Interface User Guide that is generic for the stand-alone Netlist Viewer views and the Chip Planner Canvas netlist views.
SmartDebug User Guide for SmartFusion2, IGLOO2, and RTG4	Describes how to use SmartDebug design debugging tools with SmartFusion2, IGLOO2, and RTG4 devices.
SmartDebug User Guide for PolarFire FPGA	Describes how to use SmartDebug design debugging tools with PolarFire devices.
SmartPower User Guide	Describes how to use SmartPower for power analysis.
SmartTime Static Timing Analyzer User Guide	Describes how to use SmartTime for timing analysis and how to set clock constraints for SmartFusion2,IGLOO2, RTG4, and PolarFire devices.
Timing Constraints Editor User Guide	Describes how to use the Timing Constraints Editor with timing constraints.
Tcl Commands Reference Guide	Lists all Tcl commands and parameters for the Microchip software tools.
SmartDesign User Guide	Explains how to use the SmartDesign tool, which iis used to instantiate, configure, and connect Microchip IPs, user-generated IPs, and custom/glue-logic HDL modules.
FlashPro Express User Guide	Contains information about how to program your devices using FlashPro Express.
Macro Library User Guide for RTG4	Provides descriptions of Microchip library elements for the Microchip RTG4 device family. Symbols, truth tables, and module counts (if appropriate) are included for all macros.
Macro Library User Guide for SmartFusion2 and IGLOO2	Provides descriptions of Microchip library elements for the Microchip SmartFusion2 and IGLOO2 device families. Symbols, truth tables, and module counts (if appropriate) are included for all macros.
Macro Library Guide for PolarFire	Provides descriptions of Microchip library elements for Microchip PolarFire device families. Symbols, truth tables, and module counts (if appropriate) are included for all macros.
PDC Commands User Guide for SmartFusion2, IGLOO2, and RTG4	Describes the PDC commands for I/O Attributes and Floorplanning constraints with SmartFusion2, IGLOO2, and RTG4 devices.
PDC Commands User Guide for PolarFire FPGA	Describes the PDC commands for I/O Attributes and Floorplanning constraints with PolarFire devices.

8. Revision History

Revision	Date	Description
F	08/22	• In section 3. Resolved Issues, added case number 00967427. • In section 4. Known Issues and Limitations, provided the workaround to use if the output of the Export SPI Flash Image tool for PolarFire or PolarFire SoC does not contain data for the STAGE3 initialization clients.
E	01/2022	In section 4. Known Issues and Limitations, added Libero issue that if the keys in key Mode Policy are disabled and key Mode policy is used, the disabled keys can continue to be used to ERASE and VERIFY bitstreams.
D	11/2021	• Added new section 1.3.9. Standalone Synthesis Flow. • In section 4. Known Issues and Limitations, added a known issue about unused I/O pins glitching at power-up when Libero SoC-generated defaults are used.
C	09/2021	Added section 1.3.4. Out-of-Context Derive Constraints Utility for Custom Flows. In section 4. Known Issues and Limitations, added two SmartDebug issues: • One about Data Rate, Link Width, and other values not appearing for a PCIe controller. • One about designs using DDR controller with FHB enabled may crash.
B	08/2021	Editorial updates and bug fixes.
A	08/2021	Initial Revision

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.

- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN:

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820