Product Change Notification / MAAN-13RNNO081

Date:

16-Aug-2023

Product Category:

General Purpose FPGAs, Radiation Tolerant FPGAs, System On Chip FPGAs

PCN Type:

Manufacturing Change

Notification Subject:

eSign #E000190819 Final Notice: Released of updated Libero SoC v2022.3 and PF_CCC v2.2.220 for selected products in the PolarFire FPGA device family, including MPFxxx and RTPF device families.

Affected CPNs:

MAAN-13RNNO081_Affected_CPN_08162023.pdf
MAAN-13RNNO081_Affected_CPN_08162023.csv

Notification Text:

Notification Body:
PCN Status: Final Notification

PCN Type: Manufacturing Change

Microchip Parts Affected: Please open one of the files found in the Affected CPNs section.
Note: For your convenience Microchip includes identical files in two formats (.pdf and .xls)

Description of Change: Released of updated Libero SoC v2022.3 and PF_CCC v2.2.220 for selected products in the PolarFire FPGA device family, including MPFxxx and RTPF device families. Refer to the PDF found in the Attachments section for additional details.

Pre and Post Change Summary:

	Pre Change	Post Change
Software / Tool	Libero SoC v2022.2 and PF_CCC core v2.2.214 where pre-existing design was opened in v2022.2 and an existing PF_CCC core was updated to v2.2.214.	Libero SoC v2022.3 or later, and PF_CCC core v2.2.220 or later.

Impacts to Data Sheet: None

Change Impact None

Reason for Change: Release updated Libero SoC v2022.3 and later to resolve issue that occurred on designs that were updated to Libero SoC v2022.2, where an existing PF_CCC core component instance was updated to v2.2.214, resulting in the PF_CCC core being re-generated with an incorrect PLL LOCK_CNT parameter value of 0x0. The latest version of Libero SoC is available for download on the webpage below:

https://www.microchip.com/en-us/products/fpgas-and-plds/fpga-and-soc-design-tools/fpga/libero-software-later-versions#Download%20Software

Change Implementation Status: Complete

Estimated First Ship Date: December 16, 2022 (date code: 2251)

Note: Please be advised that after the estimated first ship date customers may receive pre and post change parts.

Time Table Summary:

	December 2022					>	August 2023				
Workweek	49	50	51	52	53		31	32	33	34	35
Final PCN Issue Date									x		
Estimated Implementation Date			x								

Method to Identify Change: Not applicable. New software release is available as defined above.

Revision History: August 16, 2023: Issued final notification.

The change described in this PCN does not alter Microchip's current regulatory compliance regarding the material content of the applicable products.

Attachments:

CN_PolarFire_FPGA_CCC_Upgrade_to_v2p2p214_LOCKCOUNT.pdf

Please contact your local Microchip sales office with questions or concerns regarding this notification.

Terms and Conditions:

If you wish to receive Microchip PCNs via email please register for our PCN email service at our PCN home page select register then fill in the required fields. You will find instructions about registering for Microchips PCN email service in the PCN FAQ section.

If you wish to change your PCN profile, including opt out, please go to the PCN home page select login and sign into your myMicrochip account. Select a profile option from the left navigation bar and make the applicable selections.

Affected Catalog Part Numbers (CPN)

MPFS250T-1FCSG536T2
MPFS250T-FCSG536T2
MPFS250TS-FCV484M
MPFS250TS-FCV784M
MPFS250TS-FC1152M
MPFS250TS-FCS536M
MPFS250T-FCVG484I
MPFS250T-1FCVG484IPP
MPFS250T-FCVG484IPP
MPFS250T-1FCVG484EPP
MPFS250T-FCVG484EPP
MPFS250T-1FCVG484I
MPFS250TL-FCVG484I
MPFS250TLS-FCVG484I
MPFS250TS-1FCVG484I
MPFS250TS-FCVG484I
MPFS250T-1FCVG484E
MPFS250T-FCVG484E
MPFS250TL-FCVG484E
MPFS250T-1FCVG784I
MPFS250T-FCVG784I
MPFS250TL-FCVG784I
MPFS250TLS-FCVG784I
MPFS250TS-1FCVG784I
MPFS250TS-FCVG784I
MPFS250T-1FCVG784E
MPFS250T-FCVG784E
MPFS250TL-FCVG784E
MPFS250T-1FCG1152IPP
MPF050T-1FCSG325T2
MPF050T-FCSG325T2
MPF050T-1FCVG484E
MPF050T-1FCVG484I
MPF050T-FCVG484E
MPF050T-FCVG484I
MPF050TL-FCVG484E
MPF050TL-FCVG484I
MPF050TS-1FCVG484I
MPF050TS-FCVG484I
MPF050TLS-FCVG484I
MPF050T-1FCSG325E
MPF050T-1FCSG325I
MPF050T-FCSG325E
MPF050T-FCSG325I
MPF050TL-FCSG325E
MPF050TL-FCSG325I

MPF050TS-1FCSG325I
MPF050TS-FCSG325I
MPF050TLS-FCSG325I
MPFS160T-1FCVG484E
MPFS160T-1FCVG484I
MPFS160T-FCVG484E
MPFS160T-FCVG484I
MPFS160TL-FCVG484E
MPFS160TL-FCVG484I
MPFS160TLS-FCVG484I
MPFS160TS-1FCVG484I
MPFS160TS-FCVG484I
MPFS160T-1FCVG784E
MPFS160T-1FCVG784I
MPFS160T-FCVG784E
MPFS160T-FCVG784I
MPFS160TL-FCVG784E
MPFS160TL-FCVG784I
MPFS160TLS-FCVG784I
MPFS160TS-1FCVG784I
MPFS160TS-FCVG784I
MPFS160T-1FCSG536E
MPFS160T-1FCSG536I
MPFS160T-FCSG536E
MPFS160T-FCSG536I
MPFS160TL-FCSG536E
MPFS160TL-FCSG536I
MPFS160TLS-FCSG536I
MPFS160TS-1FCSG536I
MPFS160TS-FCSG536I
RTPF500T-1CB1509MS
RTPF500T-1CB1509PROTO
RTPF500T-1CG1509EX259
RTPF500T-1CG1509EX3
RTPF500T-1CG1509MS
RTPF500T-1CG1509PROTO
RTPF500T-1LG1509MS
RTPF500T-1LG1509PROTO
RTPF500T-CB1509PROTO
RTPF500T-CG1509PROTO
RTPF500T-LG1509PROTO
RTPF500TL-CB1509ES
RTPF500TL-CB1509PROTO
RTPF500TL-CG1509E
RTPF500TL-CG1509EX519
RTPF500TL-CG1509PROTO
RTPF500TL-LG1509ES
RTPF500TL-LG1509PROTO
RTPF500TLS-CB1509ES

RTPF500TLS-CB1509PROTO
RTPF500TLS-CG1509PROTO
RTPF500TLS-LG1509ES
RTPF500TLS-LG1509PROTO
RTPF500TS-1CB1509ES
RTPF500TS-1CB1509PROTO
RTPF500TS-1CG1509BX3
RTPF500TS-1CG1509EX155
RTPF500TS-1CG1509PROTO
RTPF500TS-1LG1509ES
RTPF500TS-1LG1509PROTO
RTPF500TS-CB1509PROTO
RTPF500TS-CG1509PROTO
RTPF500TS-LG1509ES
RTPF500TS-LG1509PROTO
RTPF500ZT-1CB1509MS
RTPF500ZT-1CB1509PROTO
RTPF500ZT-1CG1509MS
RTPF500ZT-1CG1509PROTO
RTPF500ZT-1LG1509MS
RTPF500ZT-1LG1509PROTO
RTPF500ZT-CB1509PROTO
RTPF500ZT-CG1509PROTO
RTPF500ZT-LG1509B
RTPF500ZT-LG1509ES
RTPF500ZT-LG1509PROTO
RTPF500ZTL-CB1509MS
RTPF500ZTL-CG1509MS
RTPF500ZTL-LG1509E
RTPF500ZTL-LG1509MS
RTPF500ZTLS-CB1509PROTO
RTPF500ZTLS-CG1509PROTO
RTPF500ZTLS-LG1509ES
RTPF500ZTLS-LG1509PROTO
RTPF500ZTS-1CB1509ES
RTPF500ZTS-1CB1509PROTO
RTPF500ZTS-1CG1509PROTO
RTPF500ZTS-1LG1509ES
RTPF500ZTS-1LG1509PROTO
RTPF500ZTS-CB1509PROTO
RTPF500ZTS-CG1509PROTO
RTPF500ZTS-LG1509ES
RTPF500ZTS-LG1509PROTO
MPF300T-FCVG484EX548
MPF300T-1FCG784E
MPF300T-1FCG784I
MPF300T-FCG784E
MPF300T-FCG784I
MPF300TL-FCG784E

MPF300TL-FCG784I
MPF300TLS-FCG784I
MPF300TS-1FCG784I
MPF300TS-FCG784I
MPF300T-FCG784ES0317
MPF300T-FCG784ES0323
MPF300T-1FCG784NE
MPF300T-1FCG784NI
MPF300T-FCG784NE
MPF300T-FCG784NI
MPF300TS-1FCG784NI
MPF300TS-FCG784NI
MPF300T-1FCG1152EX3
MPF300T-1FCG1152E
MPF300T-1FCG1152I
MPF300T-FCG1152E
MPF300T-FCG1152I
MPF300TL-FCG1152E
MPF300TL-FCG1152I
MPF300TLS-FCG1152I
MPF300TS-1FCG1152I
MPF300TS-FCG1152I
MPF300T-1FCG1152IX45
MPF300T-FCG1152ES0311
MPF300T-FCG1152ES0324
MPF300T-1FCSG536E
MPF300T-1FCSG536I
MPF300T-FCSG536E
MPF300T-FCSG536I
MPF300TL-FCSG536E
MPF300TL-FCSG536I
MPF300TLS-FCSG536I
MPF300TS-1FCSG536I
MPF300TS-FCSG536I
MPF500TS-FC784M
MPF500TS-FC1152MX167
MPF500TS-FC1152M
MPF500TS-FC1152MX3
MPF500T-1FCG784E
MPF500T-1FCG784I
MPF500T-FCG784E
MPF500T-FCG784I
MPF500TL-FCG784E
MPF500TL-FCG784I
MPF500TLS-FCG784I
MPF500TS-1FCG784I
MPF500TS-FCG784I
MPF500T-1FCG1152E
MPF500T-1FCG1152I

MPF500T-FCG1152E
MPF500T-FCG1152I
MPF500TL-FCG1152E
MPF500TL-FCG1152I
MPF500TLS-FCG1152I
MPF500TS-1FCG1152I
MPF500TS-FCG1152I
MPF500T-1FCG1152IX533
MPF100T-1FCVG484T2
MPF100T-FCVG484T2
MPF100T-1FCSG325T2
MPF100T-FCSG325T2
MPF100T-1FCG484T2
MPF100T-FCG484T2
MPF100T-1FCG484E
MPF100T-1FCG484I
MPF100T-FCG484E
MPF100T-FCG484I
MPF100TL-FCG484E
MPF100TL-FCG484I
MPF100TLS-FCG484I
MPF100TS-1FCG484I
MPF100TS-FCG484I
MPF100T-1FCVG484E
MPF100T-1FCVG484I
MPF100T-FCVG484E
MPF100T-FCVG484I
MPF100TL-FCVG484E
MPF100TL-FCVG484I
MPF300T-1FCVG484T2
MPF300T-FCVG484T2
MPF300T-1FCSG536T2
MPF300T-FCSG536T2
MPF300TS-FC484M
MPF300T-FCG484X547
MPF300TS-FCV484M
MPF300TS-FC784M
MPF300TS-FCS536M
MPF300T-1FCG484E
MPF300T-1FCG484I
MPF300T-FCG484EX52
MPF300T-FCG484E
MPF300T-FCG484I
MPF300TL-FCG484E
MPF300TL-FCG484I
MPF300TLS-FCG484I
MPF300TS-1FCG484I
MPF300TS-FCG484I
MPF300T-1FCG484IX548

MPF300T-1FCG484IS0322
MPF300T-1FCVG484E
MPF300T-1FCVG484I
MPF300T-FCVG484E
MPF300T-FCVG484I
MPF300TL-FCVG484E
MPF300TL-FCVG484I
MPF300TLS-FCVG484I
MPF300TS-1FCVG484I
MPF300TS-FCVG484I
MPF100TLS-FCVG484I
MPF100TS-1FCVG484I
MPF100TS-FCVG484I
MPF100T-1FCSG325E
MPF100T-1FCSG325I
MPF100T-FCSG325EX560
MPF100T-FCSG325E
MPF100T-FCSG325I
MPF100T-1FCSG325EX561
MPF100TL-FCSG325E
MPF100TL-FCSG325I
MPF100TLS-FCSG325I
MPF100TS-1FCSG325I
MPF100TS-FCSG325I
MPF100TL-FCSG325EQ347
MPF200T-1FCVG484T2
MPF200T-FCVG484T2
MPF200T-1FCSG325T2
MPF200T-FCSG325T2
MPF200T-1FCSG536T2
MPF200T-FCSG536T2
MPF200T-1FCG484T2
MPF200T-FCG484T2
MPF200TS-FCS325M
MPF200T-1FCG484E
MPF200T-1FCG484I
MPF200T-FCG484E
MPF200T-FCG484I
MPF200TL-FCG484E
MPF200TL-FCG484I
MPF200TLS-FCG484I
MPF200TS-1FCG484I
MPF200TS-FCG484I
MPF200T-FCG484EZ330
MPF200T-FCG484ES0304
MPF200T-FCG484ES0305
MPF200T-FCG484ES0306
MPF200T-1FCVG484E
MPF200T-1FCVG484I

MPF200T-FCVG484E
MPF200T-FCVG484I
MPF200TL-FCVG484E
MPF200TL-FCVG484I
MPF200TLS-FCVG484I
MPF200TS-1FCVG484I
MPF200TS-FCVG484I
MPF200T-1FCVG484ES0302
MPF200T-FCVG484ES0307
MPF200T-FCVG484ES0308
MPF200T-FCVG484ES0309
MPF200T-FCVG484ES0010
MPF200T-FCVG484ES0314
MPF200T-FCVG484IS0315
MPF200T-FCVG484IS0316
MPF200T-FCVG484ES0319
MPF200T-FCVG484ES0318
MPF200T-FCVG484ES0320
MPF200T-FCVG484IS0321
MPF200T-FCVG484ES0327
MPF200T-FCVG484ES0328
MPF200T-FCVG484ES0329
MPF200T-1FCG784E
MPF200T-1FCG784I
MPF200T-FCG784E
MPF200T-FCG784I
MPF200TL-FCG784E
MPF200TL-FCG784I
MPF200TLS-FCG784I
MPF200TS-1FCG784I
MPF200TS-FCG784I
MPF200T-FCG784EH701
MPF200T-FCG784ES0301
MPF200T-FCG784IS0303
MPF200T-FCG784ES0039
MPF200T-FCG784ES0312
MPF200T-FCG784IS0313
MPF200T-FCG784IS0325
MPF200T-FCG784IS0326
MPF200T-1FCSG325E
MPF200T-1FCSG325I
MPF200T-FCSG325E
MPF200T-FCSG325I
MPF200TL-FCSG325E
MPF200TL-FCSG325I
MPF200TLS-FCSG325I
MPF200TS-1FCSG325I
MPF200TS-FCSG325I
MPF200T-1FCSG536E

MPF200T-1FCSG536I
MPF200T-FCSG536E
MPF200T-FCSG536I
MPF200TL-FCSG536E
MPF200TL-FCSG536I
MPF200TLS-FCSG536I
MPF200TS-1FCSG536IQ302
MPF200TS-1FCSG536I
MPF200TS-FCSG536I
MPFS250T-1FCVG484T2
MPFS250T-FCVG484T2
MPFS250T-1FCVG784T2
MPFS250T-FCVG784T2
MPFS250T-FCG1152IPP
MPFS250T-FCG1152I
MPFS250T-1FCG1152EPP
MPFS250T-FCG1152EPP
MPFS250T-1FCG1152I
MPFS250TL-FCG1152I
MPFS250TLS-FCG1152I
MPFS250TS-1FCG1152I
MPFS250TS-FCG1152I
MPFS250T-1FCG1152E
MPFS250T-FCG1152E
MPFS250TL-FCG1152E
MPFS250TS-1FCG1152IPP
MPFS250T-1FCG1152IX259
MPFS250T-1FCSG536I
MPFS250T-FCSG536I
MPFS250TL-FCSG536I
MPFS250TLS-FCSG536I
MPFS250TS-1FCSG536I
MPFS250TS-FCSG536I
MPFS250T-1FCSG536E
MPFS250T-FCSG536E
MPFS250TL-FCSG536E
MPFS250T-FCSG536IPP
MPFS025T-1FCVG484E
MPFS025T-1FCVG484I
MPFS025T-FCVG484E
MPFS025T-FCVG484I
MPFS025TL-FCVG484E
MPFS025TL-FCVG484I
MPFS025TLS-FCVG484I
MPFS025TS-1FCVG484I
MPFS025TS-FCVG484I
MPFS025T-1FCSG325E
MPFS025T-1FCSG325I
MPFS025T-FCSG325E

MPFS025T-FCSG325I
MPFS025TL-FCSG325E
MPFS025TL-FCSG325I
MPFS025TLS-FCSG325I
MPFS025TS-1FCSG325I
MPFS025TS-FCSG325I
MPFS095T-1FCSG325E
MPFS095T-1FCSG325I
MPFS095T-FCSG325E
MPFS095T-FCSG325I
MPFS095TL-FCSG325E
MPFS095TL-FCSG325I
MPFS095TLS-FCSG325I
MPFS095TS-1FCSG325I
MPFS095TS-FCSG325I
MPFS095T-1FCVG484E
MPFS095T-1FCVG484I
MPFS095T-FCVG484E
MPFS095T-FCVG484I
MPFS095TL-FCVG484E
MPFS095TL-FCVG484I
MPFS095TLS-FCVG484I
MPFS095TS-1FCVG484I
MPFS095TS-FCVG484I
MPFS095T-1FCVG784E
MPFS095T-1FCVG784I
MPFS095T-FCVG784E
MPFS095T-FCVG784I
MPFS095TL-FCVG784E
MPFS095TL-FCVG784I
MPFS095TLS-FCVG784I
MPFS095TS-1FCVG784I
MPFS095TS-FCVG784I
MPFS095T-1FCSG536E
MPFS095T-1FCSG536I
MPFS095T-FCSG536E
MPFS095T-FCSG536I
MPFS095TL-FCSG536E
MPFS095TL-FCSG536I
MPFS095TLS-FCSG536I
MPFS095TS-1FCSG536I
MPFS095TS-FCSG536I
MPF050T-1FCVG484T2
MPF050T-FCVG484T2

Customer Notification (CN)

Subject: PolarFire® FPGA, PolarFire SoC, and RT PolarFire FPGA PF_CCC Core Version Update to v2.2.214 Uses Incorrect LOCK_CNT Parameter Value

May 2023

Description:

This customer notification applies to PolarFire, RT PolarFire, and PolarFire SoC designs that were upgraded to Libero SoC version v2022.2 from a prior version, where an existing instance of the PF_CCC IP core was upgraded to v2.2.214. For such designs, when the PF_CCC component version is updated to v2.2.214, the LOCK_CNT parameter value is incorrectly set to 0x0 by the PF_CCC configurator, instead of being set to 0x8. The LOCK_CNT parameter affects the amount of time a PolarFire fabric PLL waits after achieving a lock, before asserting the PLL LOCK output signal. This issue was introduced in Libero SoC v2022.2 along with PF_CCC core version 2.2.214. This issue has been fixed in Libero SoC v2022.3 and later along with PF_CCC core v2.2.220 and later.

Reason for Change:

The PF_CCC core is used to instantiate fabric clock conditioning circuitry (CCC), including fabric phase locked loop circuits (PLLs). When the PLL acquires lock onto a reference clock at its Phase and Frequency Detector (PFD) input, it normally waits for a number of extra PFD clock cycles to elapse before asserting the LOCK output signal to the user design, ensuring that a stable lock has been achieved. The PF_CCC LOCK_CNT parameter controls the number of extra PFD cycles to wait before asserting the LOCK output. The lock count is defined as 2^{LOCK_CNT} PFD cycles. For the PF_CCC core, the LOCK_CNT value should be set to 0x8, per the parameters used in the device datasheet, which adds 2^8 = 256 extra PFD cycles before asserting the PLL LOCK output.

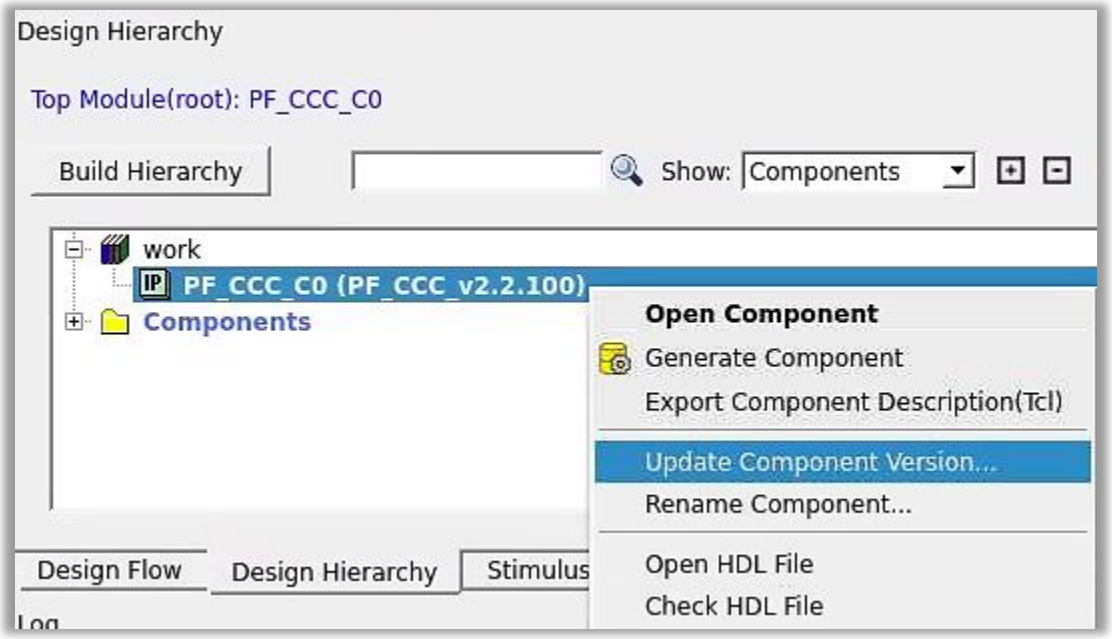
However, existing designs that were updated to Libero SoC version v2022.2, where an existing PF_CCC component instance was updated to v2.2.214, resulted in the PF_CCC core being re-generated with a LOCK_CNT parameter set to 0x0. This incorrect parameter setting reduces the number of extra PFD cycles to wait before asserting the PLL LOCK output from 256 cycles down to 2^0 = 1 cycle.

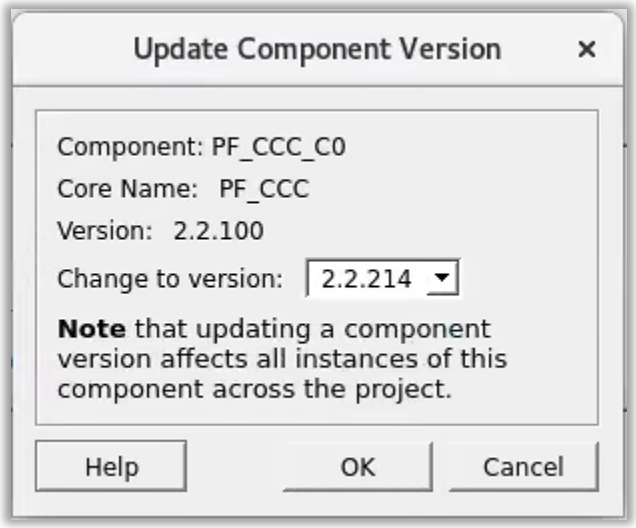
In the PF_CCC Register Map, LOCKCOUNTSEL is 4-bit wide field at offset 9 of the PLL_CTRL2 register within the memory space of each PF_CCC. The LOCK_CNT parameter value of each PF_CCC instance in the user design maps to the PF_CCC LOCKCOUNTSEL field of the PLL_CTRL2 register. The figures below are excerpts from the PF_CCC register map, describing LOCKCOUNTSEL and its impact on the LOCK output.

Reg #	Reg Offset (Hex)	Register Name	Register Description	Field Name	Width	Field Offset	User	Silicon Default Value prior to Libero Customization	Field Description
6	18	PLL_CTRL2	PLL control register	LOCKCOUNTSEL	4	9	RW	0x8	Lock count select Selects the number of PFD edges after the last cycle slip before the lock signal goes high Count is defined as 2^LOCKCOUNTSEL (e.g. with LOCKCOUNTSEL=4'd8, LOCK will go high 256 PFD periods after the last cycle slip)

Reg #	Reg Offset (Hex)	Register Name	Register Description	Field Name	Width	Field Offset	User	Silicon Default Value prior to Libero Customization	Field Description
1	4	PLL_CTRL	PLL control register	LOCK	1	25	RO		Lock detect output LOCK goes high 2^LOCKCOUNTSEL PFD cycles after the last cycle slip LOCK goes low if two cycle slips are detected within 2^LOCKCOUNTSEL PFD cycles

The “Update Component Version” action for a configured core component in Libero SoC is accessed by right-clicking the configured core component in the Design Hierarchy pane, as shown in the figures below. The “Update Component Version” action can also be accessed when the instantiated PF_CCC component core is instantiated on a SmartDesign canvas, via a right-click on the PF_CCC canvas instance. Running the “Update Component Version” action automatically regenerates the existing configured core, with the goal of using the updated core configurator while retaining the existing user configuration.





In Libero SoC v2022.3 or later, and PF_CCC core v2.2.220 or later, the incorrect “Update Component Version” behavior is fixed. Upgrading an existing design to Libero SoC v2022.3, and updating any existing PF_CCC configured core component instance to v2.2.220, correctly re-generates the PF_CCC instance with the expected LOCK_CNT parameter value of 0x8.

Application Impact:

The following design configurations are NOT impacted:

- New designs created with Libero SoC v2022.2, or later.

- Designs instantiating PF_CCC core v2.2.214, or later, from scratch, without a component version update.
- Existing designs upgraded to Libero SoC v2022.2 that do not instantiate the PF_CCC core.
- Existing designs upgraded to Libero SoC v2022.2 that do not update the component version of existing PF_CCC configured core components to v2.2.214.

Pre-existing or ongoing designs that were updated to Libero SoC version v2022.2, where existing PF_CCC core component instance(s) were updated to v2.2.214, will use a PLL LOCK_CNT value of 0x0. Using a PLL LOCK_CNT value less than 0x8 impacts the stability of the PLL LOCK output signal and could prevent the PLL LOCK output from being asserted high during design operation.

Required Action:

Existing, ongoing, or completed designs that were updated to Libero SoC version v2022.2 and an existing PF_CCC configured core component was updated to v2.2.214 can confirm whether this issue exists in the design using the following steps:

- Use the Libero Design Flow pane to run the design flow steps: 'Generate Design Initialization Data' and 'Export Design Initialization Data and Memory Report'.
- Inspect the Configured Value of PLL_CTRL2 register (of every CCC instance used in design) starting at bit offset 9. The Configured Value of PLL_CTRL2[12:9] must be 0x8. A value below 0x8 confirms the presence of the issue.

For example, in the table below, bits 9 through 12 of PLL_CTRL2 are observed to be 4'b0000. Therefore, this configuration is affected by this issue and uses the incorrect LOCK_CNT value.

PLL_SW_0 (PF_CCC_C1_0/PF_CCC_C1_0/pll_inst_0)

Register Name	Address	Silicon Default Value	Configured Value	Modified	Lock Value(*)
SOFT_RESET	0x8400000	0x00000000	0x00000000	No	N/A
PLL_CTRL	0x8400004	0x0000107C	0x0000100F	Yes	N/A
PLL_REF_FB	0x8400008	0x00000000	0x00000509	Yes	N/A
PLL_FRACN	0x840000C	0x00000000	0x00000000	No	N/A
PLL_DIV_0_1	0x8400010	0x00000000	0x01000100	Yes	N/A
PLL_DIV_2_3	0x8400014	0x00000000	0x01000100	Yes	N/A
PLL_CTRL2	0x8400018	0x00001006	0x00000036	Yes	N/A
PLL_CAL	0x840001C	0x00000000	0x0000000D	Yes	N/A
PLL_PHADJ	0x8400020	0x00004001	0x00004000	Yes	N/A

In contrast, the table below shows that bits 9 through 12 of PLL_CTRL2 are 4'b1000. Therefore, this configuration is not affected by this issue.

PLL_SW_0 (PF_CCC_C1_0/PF_CCC_C1_0/pll_inst_0)

Register Name	Address	Silicon Default Value	Configured Value	Modified	Lock Value(*)
SOFT_RESET	0x8400000	0x00000000	0x00000000	No	N/A
PLL_CTRL	0x8400004	0x0000107C	0x0000100F	Yes	N/A
PLL_REF_FB	0x8400008	0x00000000	0x00000509	Yes	N/A
PLL_FRACN	0x840000C	0x00000000	0x00000000	No	N/A
PLL_DIV_0_1	0x8400010	0x00000000	0x01000100	Yes	N/A
PLL_DIV_2_3	0x8400014	0x00000000	0x01000100	Yes	N/A
PLL_CTRL2	0x8400018	0x00001006	0x00001036	Yes	N/A
PLL_CAL	0x840001C	0x00000000	0x0000000D	Yes	N/A
PLL_PHADJ	0x8400020	0x00004001	0x00004000	Yes	N/A

Designs affected by this issue:

Use one of the following 3 methods to resolve the issue:

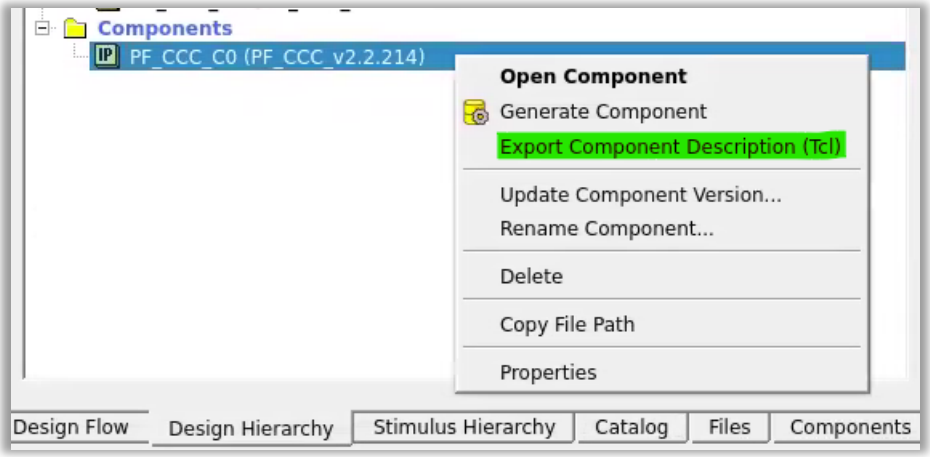
Method #1:

- Upgrade to Libero SoC v2022.3, or later, and upgrade all PF_CCC instances to v2.2.220, or later.
- Then re-run the design flow.

OR

Method #2:

- In the Libero SoC v2022.2 design, use the Design Hierarchy pane to expand the design's Components list.
- Right-click each configured PF_CCC core component and select Export Component Description (Tcl) to save a separate Tcl script for each PF_CCC core configuration used in the design.



- Then, right-click each PF_CCC component and delete it from the Design Hierarchy
- Use a text editor to update all PLL_LOCK_COUNT_0 and PLL_LOCK_COUNT_1 values from 0 to 8 in each Tcl script and save the changes. The updated Tcl script file should look like the figure below:

- Use the Libero SoC Project menu to select “Execute Script…” and run the Tcl script for each PF_CCC component in the design.
- If PF_CCC was instantiated in any SmartDesign canvas component(s), re-connect the original connections, and regenerate the SmartDesign component(s).
- Re-run the design flow and review the updated Design Initialization Data and Memory Report to confirm that the configured value of PLL_CTRL2[12:9] is 0x8.

OR

Method #3:

- In the Libero SoC v2022.2 design, delete each affected PF_CCC component from the Design Hierarchy pane and from all SmartDesign component canvas instances, if SmartDesign was used.
- Instantiate the PF_CCC core from scratch using the IP Catalog.
- Configure each new component instance using the settings that were originally used.
- If PF_CCC was instantiated via SmartDesign canvas, re-connect the original canvas connections, and regenerate the SmartDesign component.
- Re-run the design flow and review the updated Design Initialization Data and Memory Report to confirm that the configured value of PLL_CTRL2[12:9] is 0x8.

Contact Information:

For any questions about this subject, contact Microchip FPGA-BU Technical Support at the web portal below: http://www.microchip.com/support

Regards,

Microsemi Corporation, a wholly owned subsidiary of Microchip Technology Inc.