PolarFire FPGAs for Safety-Critical Applications

AN5549

Introduction (Ask a Question)

PolarFire® FPGAs are designed to address the high-reliability requirements of safety-critical systems in industrial, aviation, military, and communication applications. This application note describes the reliability features of PolarFire FPGAs.

Table of Contents

1. PolarFire FPGA Safety-Critical Features (Ask a Question)

The PolarFire FPGAs supports the following reliability features for implementing safety-critical applications.

- Single Event Effects (SEE) immune FPGA configuration
- No external configuration device required
- On-chip memories with built-in error detection and correction capabilities
- Built-in self-test
- Passivation and monitoring of unused hard IP blocks

This section provides the detailed information about the preceding features.

1.1 Reliability of FPGA Configuration Cell (Ask a Question)

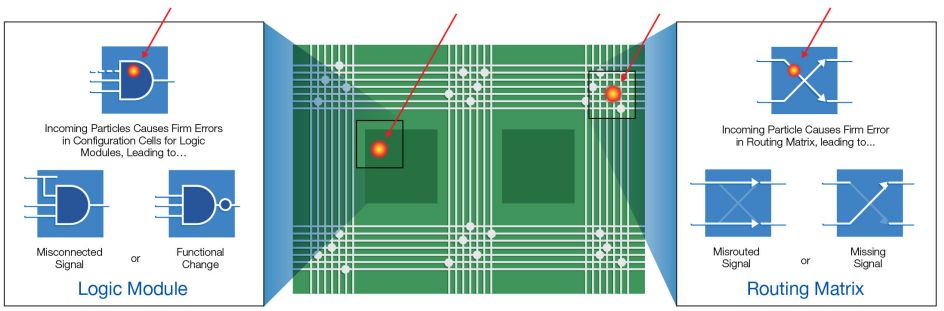
PolarFire FPGA family uses Silicon-Oxide-Nitride-Oxide-Silicon (SONOS) Non-Volatile (NV) technology to build the FPGA configuration cell. The SONOS NV technology uses a push-pull cell containing an N-channel and a P-channel NV devices. For reliability test results of PolarFire FPGAs, see Microchip FPGA and SoC Products Reliability Report.

1.2 Single Event Effects Immunity of FPGA Configuration Cell (Ask a Question)

Malfunctions in Integrated Circuits (ICs) due to radiation effects (single event effects) from high energy neutrons at ground level and high altitudes are a major concern for safety-critical applications.

Configuration upsets in FPGAs are problematic because the configuration memory must remain static and error free during all the operating hours of the device for correct operation. Any upset will be persistent until the device is powered-down or the cell is reprogrammed correctly. If an upset occurs in the erroneous state, the logic or routing of the FPGA fabric will be wrong, potentially causing not just a single wrong data value, but a string of wrong results until it is fixed. This may require a full system reboot.

Figure 1-1. Configuration Upsets in SRAM FPGAs due to Single Event Effects

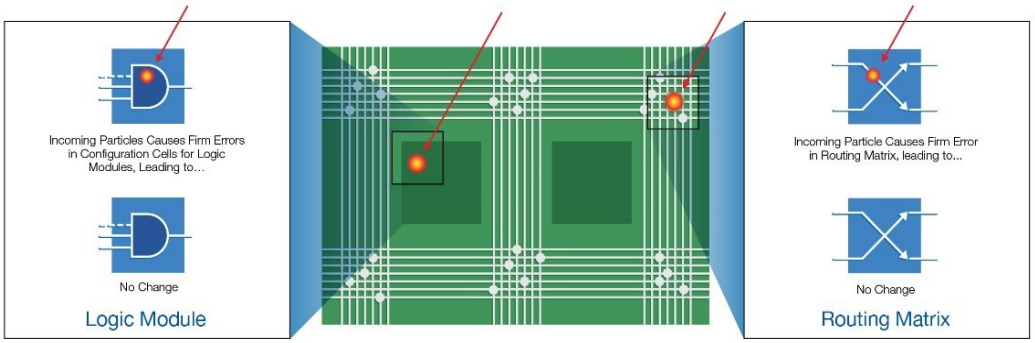


Attempts to mitigate configuration upsets in SRAM FPGAs are extremely complex. Typically, they require dual-redundant FPGAs with an external controller, which periodically searches for configuration errors in each SRAM FPGA and initiates a failover from primary to secondary FPGA while reprogramming of the primary FPGA takes place. This presumes that the system will not be detrimentally affected by the bad data produced by the FPGA during the period that the configuration SEU is undetected. It also presumes that the system can tolerate the subsequent loss of processing while the failover from primary to secondary takes place.

The PolarFire FPGA configuration is SEE immune because of the non-volatile technology, unlike the configuration memory in SRAM-based FPGAs, which can flip state due to neutron hits. In

the PolarFire FPGA family, the SONOS NV charge is stored in the nitride dielectric, which is not susceptible to charge loss from neutron hit, making it immune to Neutron induced configuration upsets.

Figure 1-2. SEE Immunity of PolarFire FPGA Configuration Cells



The first phase of neutron testing is performed on PolarFire FPGA family. The main objective is to test the product for latch-up behavior and to get soft error data on the fabric. For test results, see TR0043: PolarFire Neutron Test Results Test Report.

For more information about radiation effects, see www.microchip.com/en-us/products/fpgas-and-plds/reliability.

1.3 The Live-at-Power-Up ("Instant On") and "Single-Chip" Features (Ask a Question)

An advantage of the NV technology is that there is no need to reload the FPGA bitstream at power-up because the FPGA configuration cell retains its state after power-down. Thus, there is no need for an external flash. This improves overall system reliability. In large systems that use many FPGAs, this can result in a significant increase in reliability. For example, in some large passenger airplanes, there can be over 1,000 FPGAs used throughout the system. Eliminating configuration devices brings a significant increase in reliability. Additionally, the use of an internal NV memory for each configuration transistor means that these devices are live at power-up. This 'instant on' capability improves system robustness since the designer does not consider the various 'complexities' that occur during power-up if some devices are not working.

1.4 Error Correction and Detection Capabilities of Embedded Block RAMs (Ask a Question)

PolarFire devices include SRAM blocks as part of the hard IP blocks and FPGA fabric. Except for the uSRAM blocks, all other embedded RAM blocks are implemented with error detection and correction (ECC) capabilities to protect them from SEU effects.

The following table lists the ECC support in embedded RAM blocks.

Table 1-1. ECC Support in Embedded Block RAMs

Block	Component	ECC
FPGA Fabric	LSRAM	Yes(1)
FPGA Fabric	uSRAM	No
User Cryptoprocessor	Code and Data RAMs	Yes

(1) For LSRAMs, the ECC operation is supported only in two-port mode with a data width of 33-bit. See UG0680: PolarFire FPGA Fabric User Guide for more information.

1.4.1 ECC Operation in LSRAMs (Ask a Question)

LSRAMs configured in two-port mode with 33-bit data width supports ECC with single-bit error correction and dual-bit error detection (SECDED) capabilities. The LSRAMs are designed with an

interleave distance of 11.52 µm (center-to-center distance) to prevent multiple bit upsets within a single word. Also, in the memory array, latch-up (SEL) is prevented by including rows of tub ties spaced no more than 8.0 m.

The ECC logic in LSRAMs generates the following flags for the user logic to take necessary action:

- **SB_CORRECT**: Asserted when a single-bit error is detected. If SB_CORRECT is set without the dual-bit error flag being asserted, the corrupted bit is corrected in read data output. The data scrubbing is not implemented in the ECC logic. The scrubbing must be implemented in the user logic if required.
- **DB_DETECT**: Asserted when a dual-bit error is detected, but not corrected. Multi-bit errors (more than two bits) produce unknown results on the flags and data outputs. If DB_DETECT is set, correction is not performed on read data output.

For more information about LSRAM ECC operation, see UG0680: PolarFire FPGA Fabric User Guide.

1.4.2 ECC Operation in User Cryptoprocessor RAMs (Ask a Question)

The User Cryptoprocessor's built-in RAMs support ECC for single bit error correction and dual bit error detection. The User Cryptoprocessor can be used in the design by instantiating PF_CRYPTO macro in the design. In the event of a correctable error, the operation of the core will not be interrupted. Uncorrectable error detection causes an immediate halt of the current operation and automatic purge of the User Cryptoprocessor. The ALARM output signal of PF_CRYPTO gets set in the event of uncorrectable error detection. The purge operation zeroizes all the internal memories. An automatic soft reset is issued after completion of the purge operation. The COMPLETE output is asserted upon completion of the purge operation. All the cryptographic operations are performed through TeraFire cryptographic application library (CAL) functions and the CALPKTrfRes function is used to complete an operation. The CALPKTrfRes function returns an error code in the event of an alarm.

For more information about CAL functions, see Athena TeraFire Cryptographic Algorithm Library (CAL) Users Guide.

1.4.3 ECC Operation in PCIe (Ask a Question)

The Single Error Correction and Double Error Detection (SECDED) error reporting counters and interrupt registers display inaccurate values when ECC is enabled (default) within the PCIe hard IP block. This functional issue in the PCIe hard IP core impacts designs by incorrectly reporting single and double error counts that did not occur.

If ECC is enabled within the PCIe subsystem, ensure that the listed actions need to be taken:

- All SECDED error count registers must be ignored
- All SECDED related buffer interrupts must be disabled through the Mask registers, and associated Interrupt
- Registers must be ignored

For more information, see PCIe SECDED Reporting Defeatured in PolarFire FPGA, PolarFire SoC FPGA, and RT PolarFire FPGA.

1.5 Built-in Self-Test (Ask a Question)

PolarFire devices have a built-in self-test mechanism that can be used (optionally) to check the reliability and security of a device automatically upon power-up, or on-demand. The contents of all the non-volatile configuration memory segments, including security keys, security settings, and the FPGA fabric configuration, plus any memory pages declared as ROM by the user (all the write-protected pages) are tested using digest check. This test provides assurance against both natural and maliciously induced failures.

Digests are used for protecting data integrity. In the factory and user security segment, each logical page contains an automatically generated digest calculated dynamically at the time of programming

the data to be written. For the FPGA fabric, the digest includes an overall value covering the data to be programmed. In addition, digests are calculated and stored for the sNVM pages marked as ROM. The digests can be verified on-demand by the user, either internally using a system service, or externally using a programming instruction. In addition, the user can automatically run digest checks on each power-up. The following section describes various options to run the digest check.

An endurance limit specifies how many times a digest check of the FPGA fabric can be run. For more information about the FPGA configuration memory endurance limits, see PolarFire FPGA Datasheet. Therefore, depending on how the system is deployed and used (for example, how often it is powered-up), the on-demand digest check may be more appropriate for testing the integrity of the FPGA fabric.

1.5.1 Power-On Reset Digest Check (Ask a Question)

PolarFire FPGA device may be configured to perform automatic digest checks while powering up the user design (after power-on reset) to check the integrity of the selected non-volatile memories. The user can specify which digest to check. If any of the selected digest checks fails, a tamper event is generated to fabric for user action. The power-on digest check can be enabled and monitored using PF_TAMPER macro.

For example, if the first-stage boot code for a soft CPU is stored in the sNVM, the power-on reset digest check could be used to automatically provide a high-level of assurance that the code had not been changed whether due to natural or malicious event, since the digest was stored.

1.5.2 On-Demand Digest Check (Ask a Question)

The on-demand digest check recalculates and compares digests of selected non-volatile memories with the stored digests. A failure of any digest results in the tamper event being triggered. The on-demand digest check is invoked by calling digest check design system service. The status of the fabric digest check must be monitored by a state machine (for example, CoreABC IP core) implemented in the fabric. After checking the status of the fabric digest check, the state machine needs to issue a design reset or device reset depending on the design requirements. For more information about system services, see System Services chapter of UG0753: PolarFire FPGA Security User Guide.

Important: The LSRAMs does not retain the user data after performing digest check on FPGA fabric.

1.5.3 Exporting Digests (Ask a Question)

The stored digests can be exported via a design system service, or the JTAG or SPI-slave interface. Read Digests service returns the stored digests. For more information about running system services, see UG0753: PolarFire FPGA Security User Guide.

1.6 Passivation and Monitoring of Unused Hard IP Blocks (Ask a Question)

This section describes how to passivate and monitor unused hard IP blocks for design assurance purposes in the safety-critical applications.

1.6.1 Device I/O (Ask a Question)

All GPIO and HSIO are configured with SEU immune flash bits. Unused user GPIO and HSIO are tristated with internal weak pull-up resistors. These I/O banks have built-in calibration circuits required for features such as ODT, drive, and slew control. The calibration process runs automatically at device power-up and results in calibration codes stored in volatile registers that are subject to SEU. I/O bank's suspected to be affected by an SEU can be manually re-calibrated using the PF_INIT_MONITOR IP module. For more information, see PolarFire FPGA and PolarFire SoC FPGA User I/O User Guide.

1.6.2 Advanced I/O capabilities (Ask a Question)

I/O banks contain many advanced features including DDR with configurable clock alignments, clock and data recovery (CDR) and various memory interfaces. There are ASIC IP building blocks in the PolarFire family of devices to support these advanced capabilities. The Libero® SoC tool automatically configures these building blocks based on the designer's configuration in the tool, removing complexities from the designer. When these advanced features are not in use, the Libero SoC tool leaves these blocks in their default configurations and tie all FPGA fabric sourced inputs to a disabled logic-level through fabric flash bits. These building blocks have block-level SEU immune, flash-based, register lock bits that can be enabled to avoid inadvertent changes to the configuration registers. This provides further assurance that the block configuration remains unchanged. For more information about registers and lock bits, see Device Configuration Report and Register Locks section.

1.6.3 CCC (Ask a Question)

Clock conditioning circuits provide PLL/DLL sourced clocking to the device. The Libero SoC tool automatically configures these IP blocks based on the designer's configuration, removing complexities from the designer. When a CCC is not in use, the Libero SoC tool leaves the CCC in its default configuration and tie all FPGA fabric sourced inputs to a disabled logic-level through fabric flash bits. The CCC has a block-level SEU immune, flash-based, register lock bit that can be enabled when the IP is unused or if the IP is used but no dynamic configuration changes are part of the user design. This provides further assurance that the block configuration remains unchanged. For more information about registers and lock bits, see Device Configuration Report and Register Locks section.

The status of the CCCs can be monitored by reading the appropriate CCC status registers using the dynamic reconfiguration interface (DRI). For information about how to use DRI for dynamic register configuration, see AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note.

Important: Note that the hard IP blocks do not require to be instantiated in the design to read the status registers using DRI. See PolarFire Device Register Map for address map of CCC control and status registers.

1.6.4 System Controller (Ask a Question)

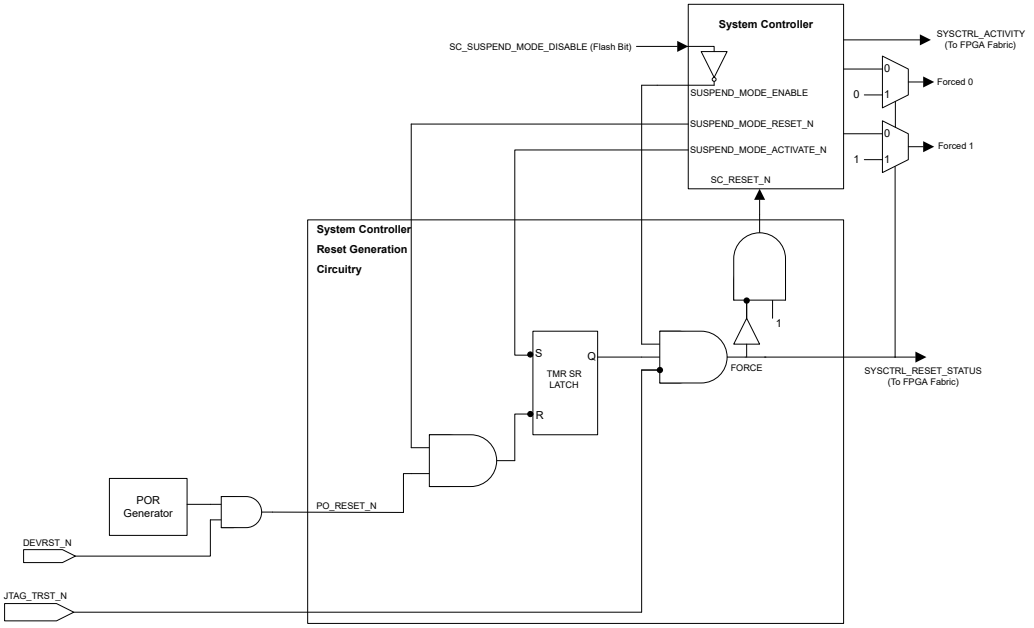
In PolarFire devices, the system controller manages device and memory initialization, programming operations, and handles the system service requests. After power-on-reset or device reset (DEVRST_N) events, the system controller performs the initialization sequence of the I/O banks, FPGA fabric, and hard IP blocks.

For high-reliability applications, such as avionics applications, the system controller must be held in suspend mode after the completion of device initialization to protect the device from unintended device programming or zeroization of the device due to SEUs.

The system controller suspend mode is designed to provide an SEU immune reset state for the system controller. The system controller reset generation circuitry is designed with a triple modular redundancy (TMR) self-refreshing latch to provide SEU immunity. In this mode, the system controller is held in reset while its output ports to the rest of the system are forced to known and well-determined states.

The following figure shows the system controller reset generation circuitry.

Figure 1-3. System Controller Suspend Mode



The following table lists the System Controller ports and description.

Table 1-2. System Controller Ports and Description

Port	Direction	Description
SUSPEND_MODE_RESET_N	Output (Internal)	Active-Low signal to reset the TMR SR Latch. Sourced from a system register.
SUSPEND_MODE_ACTIVATE_N	Output (Internal)	Active-low signal to set the TMR SR Latch. Sourced from a system register.
SUSPEND_MODE_ENABLE	Output (Internal)	Enable signal for suspend mode. Sourced from device configuration flash bit.
SC_RESET_N	Input (Internal)	System controller reset signal. This is activated after the FORCE signal.
FORCE	Input (Internal)	Indicates that all the outputs must be switched to suspend mode.
SYSCTRL_RESET_STATUS (SUSPEND_EN)	Output (To FPGA Fabric)	Direct connection of FORCE signal to the FPGA fabric indicates that FORCE is asserted, and the system controller is in suspend mode. If SYSCTRL_RESET_STATUS = 1, the system controller suspend mode is enabled. If SYSCTRL_RESET_STATUS = 0, the system controller suspend mode is disabled.
SYSCTRL_ACTIVITY (ACTIVE)	Output (To FPGA Fabric)	Signal to the FPGA fabric that represents the logical OR of the System Controller HTRANS signals. This must always be low when the system controller is in suspend mode. When not in suspend mode, this signal toggles at a variable frequency.

System controller suspend mode is controlled by a flash bit (SC_SUSPEND_MODE_DISABLE), which is set during device programming, and is not accessible either by external pin or from within the design. It is only accessed by the programming file loaded into the device, during programming. Since the SC_SUSPEND_MODE_DISABLE control bit is stored as a flash cell, it is immune to SEUs.

- If SC_SUSPEND_MODE_DISABLE = 1, the system controller suspend mode is disabled.
- If SC_SUSPEND_MODE_DISABLE = 0, the system controller suspend mode is enabled.

The suspend mode will be activated if enabled by the factory flash bit (SC_SUSPEND_MODE_DISABLE = 0) and the external JTAG reset is active. The system controller becomes active if the device is power-cycled or if a device reset (DEVRST_N) is applied, but it returns to suspend mode after the initialization sequence is completed. To restore normal operation, the device must be reprogrammed with the system controller suspend mode turned off (SC_SUSPEND_MODE_DISABLE = 1).

After the device has entered the suspend mode, the system controller is held in reset and cannot provide any system services and reprogramming services. For a full listing of device feature availability in suspend mode and for more information of system controller operation in suspend mode, see PolarFire Family System Services User Guide.

To facilitate reprogramming of the device, the JTAG_TRST_N pin is used to gate the internal FORCE signal and releases the system controller from reset. In a safety-critical environment, JTAG_TRST_N must be asserted low to prevent JTAG circuitry from affecting the I/Os due to SEUs. Releasing JTAG_TRST_N pulls the system controller out of reset and allows the device to be reprogrammed. When a programming mode instruction is loaded, the system controller sends a pulse on SUSPEND_MODE_RESET_N to clear the TMR latch so that the device can re-execute a normal boot sequence after programming is completed. Reprogramming via the system controller SPI (SC_SPI) interface is also possible. The external host must control JTAG_TRST_N. Asserting JTAG_TRST_N = 1 restores only JTAG and SPI SLAVE programming modes. All other features disabled by system controller suspend mode remain disabled.

The state of the system controller can be monitored by the FPGA fabric logic by reading the state of the SYSCTRL_RESET_STATUS (SUSPEND_EN) signal and SYSCTRL_ACTIVITY (ACTIVE) signal. Libero software provides a macro (SC_STATUS) for system controller status monitoring from fabric logic.

Confirmation of the state of the system controller suspend bit in the user design can be obtained by reviewing the `Design_Initialization_Data_Memories_Configuration_Report` generated by the Libero SoC tool.

System controller suspend mode can be enabled in the following two methods:

- Libero SoC Design tool GUI: **Project** > **Project Settings** > **Device Settings**
- Design `.tcl` file: add `-adv_options {SYSTEM_CONTROLLER_SUSPEN_MODE:1}` to the "set_device" tcl command

If the device is programmed with System Controller suspend mode enabled, the System Controller enters into suspend mode after completing device initialization (after DEVICE_INIT_DONE and AUTOCALIB_DONE gets asserted). The state of the system controller controlled PF_INIT_MONITOR outputs can be preserved during system controller suspend mode by enabling the "Latch system controller outputs" option in the PF_INIT_MONITOR IP configuration GUI.

1.6.5 PF_INIT_MONITOR (Ask a Question)

The PF_INIT_MONITOR is an IP block used to monitor the status of various steps and processes that occur during device bring up. The IP is recommended to always be instantiated in a user design to monitor the status of chip bring-up and to use the output(s) as part of the user design reset scheme. When using the system controller suspend mode feature of the device, this IP must be instantiated in the design and the IP configuration option 'Latch System Controller outputs' enabled. In this configuration, PF_INIT_MONITOR requires a clock to be supplied as an input. This clock must be sourced from the internal 160 MHz RC oscillator. This ensures all PF_INIT_MONITOR outputs maintain their state when system controller suspend mode is entered and assures that all system controller TMR circuitry is fully operational.

Important: For PolarFire® and RT PolarFire devices only, when the system controller is forced out of suspend mode, by asserting JTAG_TRST_N = 1, the outputs of the PF_INIT_MONITOR macro will be forced = 0. Since the outputs of this macro are recommended to be used for resets to user logic design, appropriate user design considerations must be made for this operational case. A power-cycle or DEVRST_N toggle is required to allow PF_INIT_MONITOR to re-assert these outputs.

1.6.6 User Cryptoprocessor (Ask a Question)

PolarFire "S" grade devices include a dedicated cryptoprocessor and NRBG (referred to as the User Cryptoprocessor) for data security applications. It provides complete support for Commercial National Security Algorithm (CNSA) suite and beyond, and includes Side-Channel Analysis (SCA) resistant cryptographic countermeasures.

The User Cryptoprocessor in the "S" grade devices can be held in reset by tying its reset and other enable signals to zero. Here the reset and enable signals are directly driven from flash configuration cells which are immune to SEUs. Users can monitor AHB-slave and AHB-master interfaces, and BUSY signal for safety-critical design assurance purposes.

The User Cryptoprocessor is disabled using a SEU immune flash cell in the non "S" grade PolarFire devices. There is no way to read the status of that flash bit during runtime. If there is a requirement to monitor the status of User Cryptoprocessor at runtime then the only way to accomplish this is by using 'S' grade device and disable the User Cryptoprocessor by holding it in reset. In "S" grade devices the state of the User Cryptoprocessor can be monitored by monitoring AHB-slave and AHB-master interfaces, and BUSY signal using fabric logic.

The following table lists the PolarFire FPGA export classification using the MPF300T as an example. The MPF100T, MPF200T, and MPF500T device densities have identical classifications. This table is applicable to extended commercial, industrial, and automotive temperature grade devices.

Table 1-3. PolarFire FPGA Export Classification

Device Options	Temperature Grade	ECCN
MPF300T/TL/TS/TLS	Extended Commercial, Industrial, and Automotive T2	5A992.c
MPF300TS	Military	3A001.a.2.c

1.6.7 Transceivers (Ask a Question)

PolarFire devices include up to 24 transceiver lanes organized in four lane quads. Each device also contains two PCIe controllers within a select quad as ASIC IP blocks. The transceiver complex consists of many sub-blocks that are interconnected via flash cell controlled interconnects and register set updates to realize a user configured design. The complexities of interconnecting these blocks and setting register values are handled automatically by the Libero SoC design tool. Unused Transceiver lanes, supporting logic blocks and PCIe IP have their register set configured into a low power, reset configuration by the Libero SoC tool. Furthermore, the Libero SoC design tool ties all fabric sourced input pins to these IP blocks to a disabled logic level through SEU immune fabric flash bits.

There are SEU immune and flash-based lock bits for each transceiver register. However, due to the interdependencies of the transceiver building blocks, it is recommended to leave the lock bits set to the unlocked state for all transceiver building block registers that are modified. This includes all quad building blocks that are named "Qx_*" and the PCIe building blocks named "PCIEx_*". Doing this ensures that the blocks get correctly configured. For more information about registers and lock bits, see Device Configuration Report and Register Locks section.

The status of these blocks can be monitored by reading the appropriate status registers using the dynamic reconfiguration interface (DRI). Only instantiation of the DRI IP is required. See

PolarFire Device Register Map for address map of Transceiver/PCIe control and status registers. For information about how to use DRI for dynamic register configuration, see AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note.

Important: The hard IP blocks do not need to be instantiated in the design to read the status registers.

1.6.8 Temperature and Voltage Monitors (Ask a Question)

PolarFire family of devices have ASIC IP temperature and voltage monitoring circuits. This IP block has only outputs to the FPGA fabric. The block is called G5_CONTROL_TVS.

The Libero SoC tool automatically configures this block based on user configuration, removing complexities from the designer. When these features are not in use, the Libero SoC tool sets the configuration registers to hold the block in reset. These blocks have SEU immune, flash based, register level lock bits that can be enabled to avoid inadvertent changes to the configuration registers. This provides further assurance that the block configuration remains unchanged. For more information about registers and lock bit, see Device Configuration Report and Register Locks section.

1.6.9 Tamper (Ask a Question)

PolarFire family of devices have an ASIC IP Tamper macro. The macro consists of the following two IP blocks:

- TAMPER_G5C
- G5_CONTROL_VOLTAGEDETECT

The Libero SoC tool automatically configures these building blocks based on the designer's configuration, removing complexities from the designer. When these features are not in use, the Libero SoC tool ties all the FPGA fabric sourced inputs to a disabled logic level through SEU immune, fabric flash bits. Libero SoC also sets the configuration registers to hold the block in reset. This macro has SEU immune, flash based, register level lock bits that can be enabled to avoid inadvertent changes to the configuration registers. This provides further assurance that the block configuration remains unchanged. For more information about registers and lock bits, see Device Configuration Report and Register Locks section.

1.6.10 SPI, System Services Interface, and uPROM (Ask a Question)

PolarFire family of devices have the additional ASIC IP blocks:

- G5_CONTROL_SPI - Interface that provides the FPGA fabric access to the external SPI interface.
- G5_CONTROL_SYS_SERVICES - System services interface to the FPGA fabric.
- uPROM - FPGA fabric interface to the uPROM.

The Libero SoC tool automatically configures these building blocks based on the designer's configuration, removing complexities from the designer. When these features are not in use, the Libero SoC tool ties all the FPGA fabric sourced inputs to a disabled logic level through SEU immune, fabric flash bits. This ensures that the IP blocks remain inoperable.

2. Device Configuration Report and Register Locks (Ask a Question)

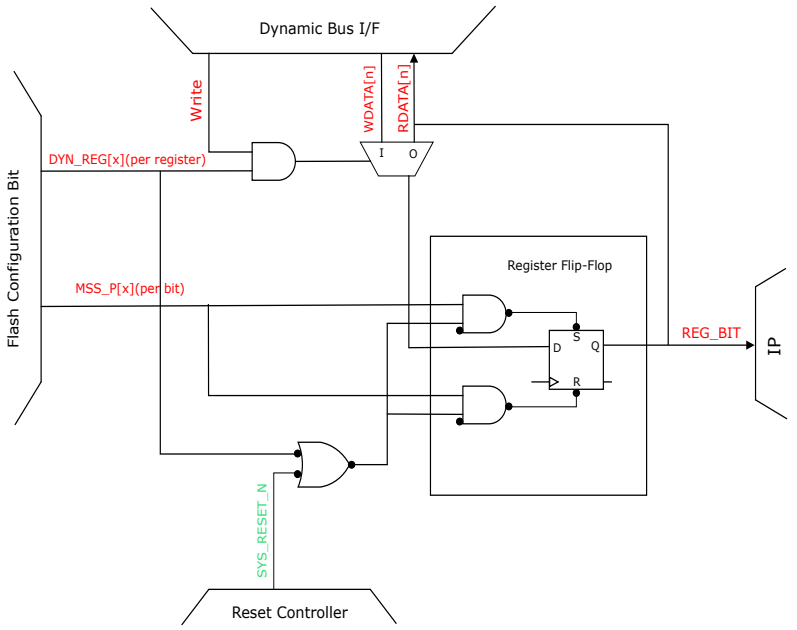
This section describes the Register Locks and Device Configuration Report.

2.1 Register Locks (Ask a Question)

The PolarFire® family of devices supports register write-protection activated through "lock" bits for the configuration registers of various IP blocks in the device. This write-protection feature enables immunity against SEUs and unintentional writes to configuration registers. Many IP blocks have flip-flop based registers which are initialized by user-configurable flash bits at power-up. After initialization, the registers content can be protected by enabling the write-protection, lock bit feature associated with the register.

The following figure shows a schematic representation of register bit protected by a lock.

Figure 2-1. Schematic Representation of Protected Register Bit



If a write-protection flash bit (DYN_REG[x]) is set, then the associated flip-flop is held in continuous reset or preset depending upon its initialization flash bit value (MSS_P[x]). Therefore, even if there is an SEU hit on the flip-flop, the flip-flop immediately goes back to the initialized flash bit value.

Some register content can be read back at run-time, using the DRI interface or an APB interface, to ensure that registers are initialized properly, and have not changed over time.

The register lock capability is provided at the register-level for many of the registers in the transceiver building blocks, user crypto block, TVS, and voltage detector block. Other IP blocks including CCCs and lane controllers have block-level register locking capability.

2.2 Device Configuration Report (Ask a Question)

The Libero SoC design tool generates `Design_Initialization_Data_Report.txt` report during the **Generate Design Initialization Data** step. This file reports the utilization of each IP block in the user design. If a block is utilized, the instance name will be reported. If the block is not utilized, it will be reported as unused. There are some user device configuration settings that can be legally changed after the **Generate Design Initialization Data** step (for example, System Controller Suspend Mode). Hence, it is recommended to run the **Export Design Initialization Data Memories Configuration Report** step once the design is completely configured and the **Generate Bitstream** step is completed. Running this step generates the

`Design_Initialization_Data_Memories_Configuration_Report.txt` file. This file includes the content of the `Design_Initialization_Data_Report.txt` file and the configuration of other design configurations, providing full information of the design configuration.

Figure 2-2. Device Configuration Report

```
Q0_PMA_LANE3 (PCIe_EP_0/PCIex4_0/PCIESS_LANE3_Pipe_AXI0)
+---------------------+-----------------+----------------+---------------------------+-------------------+-----------------+
| Register Name       | Register Address | Register Reset | Register Configured Value | Register Modified | Lock Value(*) |
+---------------------+-----------------+----------------+---------------------------+-------------------+-----------------+
| SOFT_RESET          | 0x1048000       | 0x00000000     | 0x00000000                | No                | 1             |
| DES_CDR_CTRL_1      | 0x1048004       | 0x00000000     | 0x00000000                | No                | N/A           |
| DES_CDR_CTRL_2      | 0x1048008       | 0x00000015     | 0x00000015                | No                | N/A           |
| DES_CDR_CTRL_3      | 0x104800C       | 0x00000000     | 0x0000014F                | Yes               | N/A           |
| DES_DFEEM_CTRL_1    | 0x1048010       | 0x00000015     | 0x00000015                | No                | N/A           |
| DES_DFEEM_CTRL_2    | 0x1048014       | 0x00000000     | 0x00000000                | No                | N/A           |
| DES_DFEEM_CTRL_3    | 0x1048018       | 0x00000000     | 0x00004F00                | Yes               | 1             |
+---------------------+-----------------+----------------+---------------------------+-------------------+-----------------+

Q2_PCS_LANE3 (Unused)
+---------------+-----------------+----------------+---------------------------+-------------------+-----------------+
| Register Name | Register Address | Register Reset | Register Configured Value | Register Modified | Lock Value(*) |
+---------------+-----------------+----------------+---------------------------+-------------------+-----------------+
| SOFT_RESET    | 0x108000        | 0x00000000     | 0x00000100                | Yes               | N/A           |
| LFWF_R0       | 0x108004        | 0x00000000     | 0x00000000                | No                | 1             |
| LOVR_R0       | 0x108008        | 0x00000010     | 0x00000000                | Yes               | 1             |
| LPIP_R0       | 0x10800C        | 0x0000005D     | 0x00000058                | Yes               | 1             |
| L64_R0        | 0x108010        | 0x01400000     | 0x01400000                | No                | N/A           |
+---------------+-----------------+----------------+---------------------------+-------------------+-----------------+
```

2.3 Recommendations for Lock Bits (Ask a Question)

Transceiver Related IP Blocks

This applies to all transceiver building blocks that are named "Qx_*" and the PCIe building blocks that are named "PCIEx_*". If the Lock Value field is equal to 1, then the register has lock capability. Only registers with the “Register Modified field = No” must be considered for locking. These registers are not modified from their default value so there is no potential to lock out the power-up/DEVRSTn UIC configuration sequence that may dynamically configure the register. Due to interdependencies of these IP blocks, if any transceiver capability is added to the user design, then apply this recommendation to all transceiver-related blocks.

Non-transceiver Related IP Blocks

If the Lock Value field is equal to 1, then the register has lock capability. If the IP block is not used in the design, (the instance name = N/A) then it is safe to lock the IP block. If the IP block is used in the design, then it is safe to lock the IP block when there is no plan/capability to dynamically change the configuration of the IP during design operation.

Block-level Locked IP Blocks

Some IP blocks have a single lock bit for all registers in the block. These IP blocks are listed in the subsection of the report titled "Block-Level Control for locking peripheral blocks".

Figure 2-3. Block-level Control For Locking Peripheral Blocks

```
Block-Level Control For Locking Peripheral Blocks.
+-------------------------+-------------------------+-----------------------------------------------+---------------+
| Peripheral Block Name   | Name For Locking The Block | Instance Name                                 | Lock Value(*) |
+-------------------------+-------------------------+-----------------------------------------------+---------------+
| CCC PLL                 | PLL_NE_0_LOCK           | PF_DDR3_SS_0/CCC_0/pll_inst_0                 | 1             |
| CCC PLL                 | PLL_NE_1_LOCK           | N/A                                           | 1             |
| CCC PLL                 | PLL_NW_0_LOCK           | N/A                                           | 1             |
| CCC PLL                 | PLL_NW_1_LOCK           | PF_DDR4_SS_0/CCC_0/pll_inst_0                 | 1             |
+-------------------------+-------------------------+-----------------------------------------------+---------------+
```

If the IP block is not used in the design, (the instance name = N/A) then it is safe to lock the IP block. If the IP block is used in the design, then it is safe to lock the IP block when there is no plan to dynamically change the configuration of the IP during design operation. For example, if the user plans to dynamically change a CCC PLL frequency while the design is operational, then this IP block cannot be locked.

2.4 Unused IP Pin Tie-offs (Ask a Question)

The final section of the device configuration report is the unused IP pin tie-offs. When an IP block is unused in a design, all fabric driven IP block inputs are connected (tied-off) to either a logic '1' or a logic '0'. This tie-off is through an SEU immune FPGA fabric flash bit. The inputs are tied-off to logic levels that places the IP block into a low-power reset state disabling the operation of the block.

This section reports all unused IP blocks with the tie-off connection of each FPGA fabric input to the block.

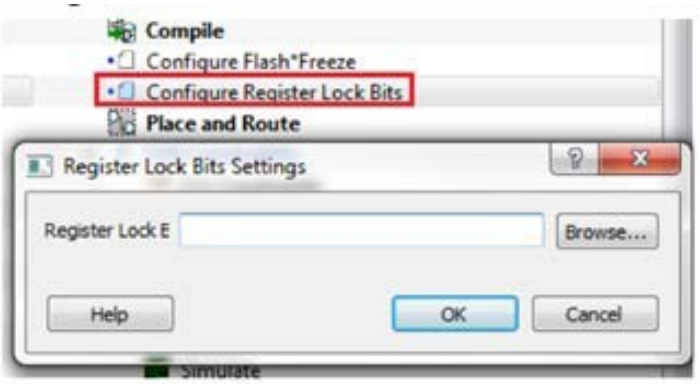
2.5 Setting Lock Bit Protection (Ask a Question)

The Configure Register Lock Bits option available in the Libero SoC Design tool is used to write-protect/lock user selected IP registers. Register lock bits are set in a text (`*.txt`) file, which the user then imports into the Libero SoC project. Use the following procedure in Libero SoC design tool to import the lock bit file into a design:

- From the **Design Flow** window, double-click **Configure Register Lock Bits** to open the configurator.
- Navigate to the text file (`*.txt`) that contains the register lock bits settings (see the following figure).

Subsequent FPGA bitstreams generated will have IP register lock bits set according to the file.

Figure 2-4. Register Lock Bit Settings



2.5.1 Lock Bit File (Ask a Question)

A default lock bit file is generated by Libero SoC when the design flow step **Generate FPGA Array Data** is executed. The default file is located at `/designer//_init_config_lock_bits.txt`.

This file contains an entry for every lock bit available in the device. The format of the lock bit file is: `<lock bit value>`.

- **lock_bit_name**: Lock bit name in the format `<Physical block name>_<register name>_LOCK`
- **lock_bit_value**: 1 register unlocked (default); 0 register locked

Rename this file and edit the settings of each register lock bit as desired. Once complete, add the file to the design flow as described in Setting Lock Bit Protection section.

Important: Changing the content of the lock-bits file does not invalidate place-and-route. However, you must regenerate the FPGA array data and regenerate the bitstream for lock bit changes to take effect. The current state of lock bits can be observed in the regenerated lock bits file that is recreated during the **Generate FPGA Array Data**.

3. Configuring SmartDebug Circuits for Safety-Critical Applications (Ask a Question)

This section describes the SmartDebug architecture and SmartDebug radiation exposure and mitigation.

3.1 SmartDebug Architecture (Ask a Question)

PolarFire devices contain additional logic to support design debug. SmartDebug circuitry supports the following debug capabilities:

- Live Probe
- Active Probe
- Fabric RAM Block access
- sNVM read access
- Transceiver and DDR debug
- Fabric Hardware Breakpoint (FHB)

The SmartDebug circuitry is mastered by the embedded System Controller, which is controlled by commands over the JTAG interface. The System Controller converts the JTAG commands to various bus accesses to control the SmartDebug logic.

3.2 SmartDebug Radiation Exposure and Mitigation (Ask a Question)

The underlying SmartDebug circuitry is implemented primarily by ASIC logic. The exception being the Fabric Hardware Breakpoints, which are implemented as FPGA fabric logic. The ASIC logic is asynchronous logic with some amount of synchronous logic. Adhering to the recommendations provided, it holds the majority of synchronous logic in reset, minimizing any deleterious effects due to SEE for safety-critical applications. The remaining asynchronous logic would require multiple SEE events to result in unwanted SmartDebug access. This is due to the bus style architecture required to access these circuits.

To minimize the impact of SEE exposure, it is recommended to disable and/or minimize the SmartDebug circuitry as follows:

1. Enable the System Controller suspend mode. In this configuration, the System Controller is driven into TMR'd asynchronous reset. In this state, the controller cannot initiate access to the SmartDebug circuity due to an SEE event. In this mode, the System Controller reset output holds the SmartDebug logic in reset.
2. Disable the Fabric Hardware Breakpoints (FHB). This removes the circuitry from the FPGA fabric, reducing SEE footprint of the design proportionally.
3. Reserve the two available Live Probe pins and do not use them as user I/O.

Any SEU's in SmartDebug circuitry are covered, by default, in Microchip SEU reports, which show overall SEU rates at levels acceptable or better for aviation applications.

4. FPGA Programming Circuitry (Ask a Question)

The underlying device programming circuitry is implemented by a dedicated programming bus. This parallel bus implements address, data and control bus parity, preventing incorrect write cycles in the event of SEU. Also, all critical configuration registers are TMR protected. When unused, this bus is held in asynchronous reset for further SEU protection.

Any SEU's in programming circuitry are covered, by default, in Microchip SEU reports, which show overall SEU rates at levels acceptable or better for aviation applications.

5. DO-254 (Ask a Question)

Microchip FPGA families have more than 20 years of proven performance across product deployments in hundreds of commercial aviation systems on Airbus, Boeing, and other aircraft.

These devices perform critical functions in Design Assurance Level (DAL) A and B applications such as flight computers, braking systems, cockpit displays, engine controls, actuator systems, safety warning systems, cabin data management, and more.

Microchip product families meet the stringent requirements of 10^6 device hours of operations, which are needed for the most safety-critical applications for DO-254 certification.

Microchip product families offer dissimilar technologies (anti-fuse, flash, SONOS), which is ideal for safety-critical and redundant systems. Microchip FPGAs address a critical high-reliability requirement for commercial aviation with zero Failure in Time (FIT) rate for FPGA configuration.

Microchip provides validation artifacts to assist system level designers with DO-254 certification. For more questions about validation artifacts, contact aviation@microchip.com.

6. IEC 61508 (Ask a Question)

Microchip is offering an IEC 61508 certified Functional Safety Data Package for the following families:

- ProASIC® 3, ProASIC3e, and ProASIC3L
- IGLOO®, IGLOOe, IGLOO nano, and IGLOO PLUS
- SmartFusion®

The functional safety packet is designed to assist with IEC 61508 certification and includes:

- Information on the relevant devices
- Libero® SoC Design Suite v11.5 SP2 certified by TUV
- Libero SoC documentation
- Relevant IP cores and associated documentation
- IEC 61508 Safety Data Manual

The packet is available for purchase using the ordering code SAFETY-PKG-G3. For more information, see www.microchip.com/en-us/solutions/technologies/functional-safety/iec-61508.

7. Revision History (Ask a Question)

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

Revision	Date	Description
A	09/2024	The following is the list of changes in revision A of the document: • The document was migrated to Microchip template. • The document number was updated to DS00005549 from AC478. • Updated ECC Operation in PCIe section with information related to PCIe SECDED reporting single and double error counts. • Removed PCIe block information from Table 1-1 section. • Updated Figure 2-1 in the Register Locks section.
4.0	—	The following is a summary of changes made in this revision. • Added information about Device I/O, Advanced I/O capabilities, and CCC. • Updated information about how to enable system controller suspend mode. See System Controller. • Added information about PolarFire Initialization Monitor IP block. See PF_INIT_MONITOR. • Added information about instantiation of User Cryptoprocessor in user design. See User Cryptoprocessor. • Added information about Transceiver IP block. See Transceivers. • Added information about temperature and voltage monitor IP block. See Temperature and Voltage Monitors. • Added information about Tamper macro and its IP blocks. See Tamper. • Added information about ASIC IP blocks. See SPI, System Services Interface, and uPROM. • Added information about register locks, configuration reports, recommendation for lock bits, unused IP pin tie-offs and setting lock bit protection. See Device Configuration Report and Register Locks. • Added information about SmartDebug architecture and its radiation exposure. See Configuring SmartDebug Circuits for Safety-Critical Applications. • Added information about FPGA Programming Circuitry.
3.0	—	The following is a summary of changes made in this revision. Information about SYSCTRL_ACTIVITY port description was updated. See Table 1-2. Information about PolarFire FPGA export classification was updated. See Table 1-3. Updated Figure 1-3.
2.0	—	Information about System Controller was updated.
1.0	—	The first publication of this document.

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**

Microchip Information

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 products:

- 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, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- 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. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, TimeCesium, TimeHub, TimePictra, TimeProvider, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic

ISBN: 978-1-6683-0295-8

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 - Hod Hasharon** Tel: 972-9-775-5100 **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