XILINX®

XAPP779 (v1.1) February 19, 2007

# Correcting Single-Event Upsets in Virtex-II Platform FPGA Configuration Memory

Authors: Brendan Bridgford, Carl Carmichael, Chen Wei Tseng

## Summary

Designers of space-based application must be concerned with the effect of single-event upsets (SEUs) on FPGA configuration memory. Changes to this memory can cause changes in the functionality and performance of the device. This application note describes the use of partial reconfiguration and readback in Virtex™-II FPGAs for the purpose of detecting and correcting single event upsets to the configuration memory array induced by cosmic rays.

It is essential for the reader to have a basic understanding of the Virtex-II SelectMap interface as well as configuration and readback operations. An in-depth review of Virtex-II user guide is strongly encouraged.

## Introduction

In-orbit, space-based, and extra-terrestrial applications must consider the effects high energy charged particles (radiation) can have on electronic components. In particular, single event upsets (SEUs) can alter the logic state of any static memory element (latch, flip-flop, or RAM cell). Since the user-programmed functionality of an FPGA depends on the data stored in millions of configuration latches within the device, an SEU in the configuration memory array can have adverse effects on the expected functionality.

A static upset in the configuration memory is not synonymous with a functional error. Upsets might have no effect on functionality. Design mitigation techniques, such as using TMRTool to implement triple module design redundancy, or triple FPGA deployment, can harden the application against single events upsets. However, the upsets must be corrected so that errors do not accumulate. For information about TMRTool, please refer to UG156, *Xilinx TMRTool User Guide.*

The Virtex-II Series FPGA SelectMap interface provides the most efficient, post-configuration read/write access to the configuration memory array. Readback is a post-configuration read operation of the configuration memory, and partial reconfiguration is a post-configuration write operation to the configuration memory. Readback and partial reconfiguration can be used to allow a system to detect and repair SEUs in the configuration memory without disrupting its operations or completely reconfiguring the FPGA.

This application note is presented with the assumption that the reader has a solid understanding of basic configuration and readback operations, bitstream format, and command structure of the Virtex-II Series configuration logic and SelectMap interface. A careful review of the latest user guide provides this information (refer to UG002, *Virtex-II Platform FPGA User Guide*).

## Configuration Memory Architecture Overview

The most basic configuration unit in the Virtex-II architecture is a frame. Groups of configuration frames compose columns such as configuration logic block (CLB), input output block (IOB), input output interconnect (IOI), global clock (GCLK), block memory (BRAM), or BRAM Interconnects columns. Columns are then sorted into three block types: CLB (includes GLKC, CLB and IOB1&2), BRAM, and BRAM Interconnects. Since a frame expands from the top to the bottom of the device, frame sizes differ in various regions of the device.

© 2006–2007 Xilinx, Inc. All rights reserved. All Xilinx trademarks, registered trademarks, patents, and further disclaimers are as listed at http://www.xilinx.com/legal.htm. PowerPC is a trademark of IBM Inc. All other trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

NOTICE OF DISCLAIMER: Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

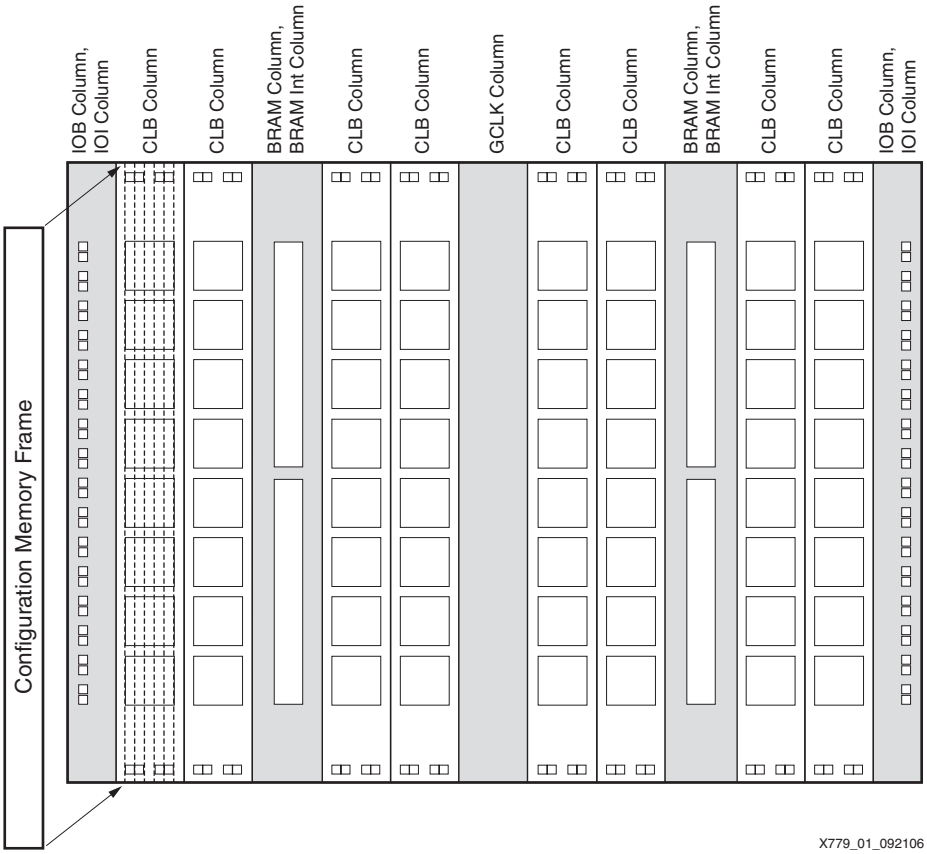


*Figure 1:* **Virtex-II Configuration Memory Architecture**

*Table 1:* **Virtex-II Frame Number per Column**

| Column Type → | IOB | | IOI | | CLB | | BRAM | | BRAM Interconnect | | GCLK | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Device** | **Columns per Device** | **Frames per Column** | **Columns per Device** | **Frames per Column** | **Columns per Device** | **Frames per Column** | **Columns per Device** | **Frames per Column** | **Columns per Device** | **Frames per Column** | **Columns per Device** | **Frames per Column** |
| **XQR2V1000** | 2 | 4 | 2 | 22 | 32 | 22 | 4 | 64 | 4 | 22 | 1 | 4 |
| **XQR2V3000** | 2 | 4 | 2 | 22 | 56 | 22 | 6 | 64 | 6 | 22 | 1 | 4 |
| **XQR2V6000** | 2 | 4 | 2 | 22 | 88 | 22 | 6 | 64 | 6 | 22 | 1 | 4 |

*Table 2:* **Virtex-II Frame Count, Frame Length, and Bitstream Size**

| Device | Frames | Frame Length (32-bit Words) | Configuration Bits(1) | Default Bitstream Size(2) |
|---|---|---|---|---|
| **XQR2V1000** | 1104 | 106 | 3,744,768 | 4,082,592 |
| **XQR2V3000** | 1804 | 166 | 9,582,848 | 10,494,368 |
| **XQR2V6000** | 2508 | 246 | 19,742,976 | 21,849,504 |

**Notes:**

1. Number of configuration bits = (Number of Frames) x (Frame Length)
2. Default bitstream size = (Configuration bits) + Overhead.

To access and perform configuration or readback operations, 32-bit data packets are issued to target different configuration registers. Configuration data packets are made up of a 32-bit type 1 header (Table 3) that can be followed by either a 32-bit data body or a 32-bit type 2 packet header (Table 4).

*Table 3:* **Type 1 Packet Header**

| Type | | | WR | RD | Register Address | | | | | | | | | | | | | | RSVD | | Word Count | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 1 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 | 0 | x | x | x | x | x | x | x | x | x | x | x |

*Table 4:* **Type 2 Packet Header**

| Type | | | WR | RD | Word Count | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 1 | 0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

*Table 5:* **Virtex-II Configuration Register Address for Type-1 Packet**

| Register Name | Read | Write | Address | Description |
|---|---|---|---|---|
| CRC | Y | Y | 00000 | CRC register |
| FAR | Y | Y | 00001 | Frame address register |
| FDRI | N | Y | 00010 | Frame data input register (write configuration data) |
| FDRO | Y | N | 00011 | Frame data output register (readback configuration data) |
| CMD | Y | Y | 00100 | Command register |
| CTL | Y | Y | 00101 | Control register |
| MASK | Y | Y | 00110 | Masking register for CTL |
| STAT | Y | N | 00111 | Status register |
| LOUT | N | Y | 01000 | Legacy output register (DOUT for daisy chain) |
| COR | Y | Y | 01001 | Configuration option register |
| MFWR | N | Y | 01010 | Multiple frame write register |
| FLR | Y | Y | 01011 | Frame length register |
| KEY | N | Y | 01100 | Initial key address register |
| CBC | N | Y | 01101 | Initial CBC value register |
| IDCODE | Y | Y | 01110 | Device ID register |

*Table 6:* **Virtex-II Command Register Codes**

| Command | Code | Description |
|---|---|---|
| WCFG | 0001 | **Write Configuration Data**. Used prior to writing configuration data to the FDRI. |
| MFWR | 0010 | **Multiple Frame Write Register**. Used to perform a write of a single frame to multiple frame addresses. |
| LFRM | 0011 | **Last Frame.** Deasserts the GHIGH_B signal, activating all interconnect. The GHIGH_B signal is asserted with the AGHIGH command. |
| RCFG | 0100 | **Read Configuration Data**. Used prior to reading configuration data from the FDRO. |
| START | 0101 | **Begin Startup Sequence.** Initiates the startup sequence. The startup sequence begins after a successful CRC check and a DESYNC command are performed. |
| RCAP | 0110 | **Reset Capture.** Resets the CAPTURE signal after performing readback-capture in single-shot mode. |
| RCRC | 0111 | **Reset CRC**. Resets the CRC register. |
| AGHIGH | 1000 | **Assert GHIGH_B Signal**. Places all interconnect in a high-Z state to prevent contention when writing new configuration data. This command is only used during shutdown reconfiguration and readback. Interconnect is reactivated with the LFRM command. |
| SWITCH | 1001 | **Switch CCLK Frequency.** Updates the frequency of the Master CCLK to the value specified by the OSCFSEL bits in the COR. |
| GRESTORE | 1010 | **Pulse the GRESTORE Signal.** Sets/resets (depending on user configuration) IOB and CLB flip-flops. |
| SHUTDOWN | 1011 | **Begin Shutdown Sequence.** Initiates the shutdown sequence, disabling the device when finished. Shutdown activates on the next successful CRC check or RCRC instruction (typically an RCRC instruction). |
| GCAPTURE | 1100 | **Pulse GCAPTURE.** Loads the capture cells with the current register states. |
| DESYNCH | 1101 | **Reset DALIGN Signal.** Used at the end of configuration to desynchronize the device. After desynchronization, all values on the configuration data pins are ignored. |

Configuring and reading back configuration frames requires the frame address to be specified. The Virtex-II frame addressing begins with `0x00000000h` which is the start of the GCLK column in the center of the device. The address then increments to target the frame at the left most side which is the IOB column and move its way to the right.

The frame address register (FAR) is divided into three main fields, block address (BA), major address (MJA), and minor address (MNA). When FDRI or FDRO commands request multiple frames to be configured or read back, FAR auto-increments the MNA, MJA, and BA accordingly. However, not all MNA and MJA fields are used. The number of minor and major addresses per column type is shown in Table 1.

*Table 7:* **Virtex-II Frame Address Composition**

| | | | | | BA | | MJA | | | | | | | | MNA | | | | | | | | Byte Number | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 | 0 | x | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Block address 00 covers all GLCK, IOB, IOI, and CLB columns while block address 01 includes all BRAM columns, and block address 10 targets all BRAM Interconnects columns.

The major address is a function of the device size and is determined by the number of columns in a device. The minor address indicates the specific frame within the column (refer to Table 1 for the number of columns and frames within a column).

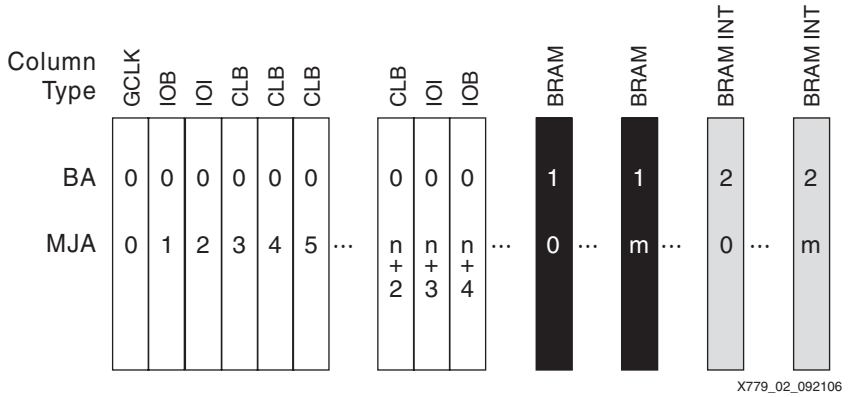


*Figure 2:* **Column-Level Configuration Memory Map for Virtex-II**

## Design Considerations for Configuration and Readback

For readback and partial reconfiguration operations, there are special considerations for design implementation. When a look-up table (LUT) is implemented as either SRL16 or distributed RAM (LUTRAM), readback or active partial reconfiguration can corrupt the contents of such primitives. Block RAM (BRAM) has the same problem as SRL16 and LUTRAM when accessed through the configuration interface while in operation. Therefore, BRAM block addresses must be avoided all together during readback and reconfiguration. However, BRAM Interconnect frames only contain configuration information and thus can be freely accessed just as CLB frames not containing special RAM primitives.

While CLB and IOB flip-flops have programmable features selected by configuration latches, the flip-flop register storage elements are separate from configuration latches and cannot be accessed through configuration. Therefore, readback and partial reconfiguration do not affect the data stored in these registers. However, if the CAPTURE_VIRTEX2 component is used, then a dedicated configuration latch updates with the current flip-flop register value when capture is triggered. This action could result in readback mismatch if this data is not masked for readback comparison.

Furthermore, for readback and reconfiguration through the SelectMap interface, the SelectMap pins need to be persisted via bitgen option.

For Virtex-II devices, the implementation of the design should ensure the Config Stepping constraint is set to 1 and bitgen's FREEZEDCI option is set to NO. When implementing a design through the ISE™ Project Navigator, set the "DCI Update Mode" to "Continuous" in the "Configuration Options" tab from the "Generate Programming File" properties. These two options ensure that only a single FDRI write command is used in the configuration bitstream, simplifying the active readback and reconfiguration process. Refer to the constraints guide and *Development System Reference Guide* for more info on Config Stepping and bitgen options.

### Single Event Upset and Functional Interrupt Detection

SEUs can induce different types of effects within an FPGA with varying levels of severity. Upsets in the configuration memory can corrupt the user design or have no effect at all. Upsets in the configuration control logic might result in a momentary loss to the configuration interface or a clearing of the entire configuration memory. Therefore, different detection methods are needed for each event type.

Most upsets within the configuration memory do not have any effect because the majority of configuration memory cells are not relevant to a typical user design. The routing control bits account for over 60% of the configuration bits while a typical design only uses approximately 10 to 20% of the routing resources. However, an upset to the utilized configuration memory bits might interrupt the user design. Readback of the configuration memory and verification of the readback data against the stored configuration data can be used to determine if any particular configuration memory bit has been upset or not, but does not automatically determine if the upset cell is relevant to the user design.

An upset to the configuration logic could result in a loss to the configuration interface, a power-on-reset (POR), or other effects, completely interrupt the device operation. Such events are referred to as SEFIs, single event function interrupts.

To quickly and properly identify SEFIs, a combination of different detection methods should be implemented. A strong detection method can quickly and accurately identify a SEFI while a weak detection method might only detect an event slowly or not at all.

## Virtex-II SEU Detection Methods

The following detection methods are devised for each known SEU and SEFI event with a unique detection characteristic. Combining these methods provides an all-encompassing solution for mitigating SEU and SEFI events. However, some applications might not employ all methods depending on the designer's intent to automatically scrub or implement a readback-detection-only circuit, or whether configuration bit errors are to be counted and tracked. While there are many options for the designer to consider, this document provides an example combining all of the following methodologies into a complete mitigation solution.

**Monitoring DONE:** After device configuration, the DONE pin should remain High. If at any time, DONE is de-asserted, indicating an upset in the power-on rest circuitry resulting in a global or partial clearing of the configuration memory, requiring a full reconfiguration for proper operation.

**Writing and reading the frame address register:** This operation should be performed before each scrubbing and readback operation and can quickly identify a SelectMAP interface problem or other configuration logic disturbance.

**Calculating and comparing a readback CRC:** When performing readback, the readback data can be used to calculate a CRC value on a frame or device basis. A golden CRC value needed for this comparison cab be calculated using a predetermined value generated during configuration from the PROM data, or generated from an initial readback after configuration verifies with no bit errors. If no bit comparison is performed, then an incorrect golden CRC values can result in a false SEFI detection. Therefore, reconfiguration should be preformed until a valid golden CRC is obtained.

**Monitoring BUSY:** During write operations, the BUSY pin should stay Low. The BUSY pin is held High during a read operation; deassertion indicates that valid readback data is available on the data bus. If BUSY remains High for more than 32 clock cycles after a transition from write to read indicates that a SMAP SEFI has occurred.

**Verify readback data:** During readback, the readback data is compared against the configuration data to ensure no differences. CRC values can be calculated from the readback data at this time. Verification operations require the non-volatile configuration memory storage device to be accessed.

**Monitoring status register:** Status register holds information on critical device status. Bit 12 is the state of DONE pin. Bit 7 indicates GHIGH status, which control the internal pull-up of signals. Bit 6 is the global write enable and bit 5 is the global 3-state status.

*Table 8:* **Virtex-II SEU Detection Methods**

| SEU Detection Method | Strong for Detecting | Weak for Detecting |
|---|---|---|
| Monitoring Done | POR SEFI | Non-SEFI SEU |
| Writing and reading a predefined value to Frame Address Register. | SelectMap SEFI | Non-SEFI SEU<br>POR SEFI |
| Calculating and comparing CRC value on the readback data | Non-SEFI SEU | POR SEFI<br>SelectMap SEFI |
| Monitoring BUSY | SelectMap SEFI | Non-SEFI SEU<br>POR SEFI |
| Read and verify readback data against configuration data | Non-SEFI SEU | POR SEFI<br>SelectMap SEFI |
| Monitoring Status Register | Global signal SEFI<br>POR SEFI | Non_SEFI SEU |

## SEU and SEFI Mitigation

There are various SEU mitigation schemes that can be combined to accommodate a particular application. Before choosing a technique to mitigate SEUs, designers should consider both the criticality of the design as well as the hostility of the environment.

Table 9 shows some example error rates for a XQR2V6000CF1144V in a geosynchronous orbit (solar quiet conditions) for the configuration memory, block memory, POR-SEFI and SMAP-SEFI. For a fully TMR design with a configuration management engine, the resulting functional device error rate is dominated by the summation of the POR and SMAP SEFI rates.

*Table 9:* **GEO Error Rates for XQR2V6000 (CREME96)**

| XQR2V6000 – 36,000 km | Mean Time to Error | Units |
|---|---|---|
| Configuration Memory | 1.8 | Hours |
| Block Memory | 11.8 | Hours |
| POR-SEFI | 221 | Years |
| SMAP-SEFI | 181 | Years |

If the application cannot afford any possible downtime or sudden disruption within the SEFI error rates for the applicable orbit, then the mitigation scheme should employ triplicating the FPGAs, a configuration management engine, plus an downstream, radiation-hardened output voter.

If the device SEFI upset rate is acceptable for the mission and the environment is harsh, the design should be triplicated with TMRTool and a configuration management engine employed to prevent SEU accumulation in the configuration memory. For more information on TMRTool, refer to http://www.xilinx.com/products/milaero/tmr/index.htm

Some applications have an inherent duty cycle as a part of normal operation, either powering down on a predetermined schedule or frequently idling of the FPGA, allowing for soft-reboots or full reconfiguration. For applications where the FPGA can be fully reconfigured at a higher rate than the configuration memory upset rate, a mitigation scheme might not be required.

***Note:*** Power-cycling is not necessary for repairing any SEU or SEFI in a Xilinx Virtex-II FPGA. All SEUs to the configuration memory can be mitigated by scrubbing. All SEFIs are recovered by pulsing the PROG pin and fully reconfiguring the FPGA. Single event transients (SETs) and SEUs in user storage elements can be mitigated through design redundancy (XTMR).

## Configuration Management Engine

For most orbital applications, a configuration management engine is required to prevent SEUs from accumulating in the configuration memory and to detect and recover from SEFIs in the configuration control logic.

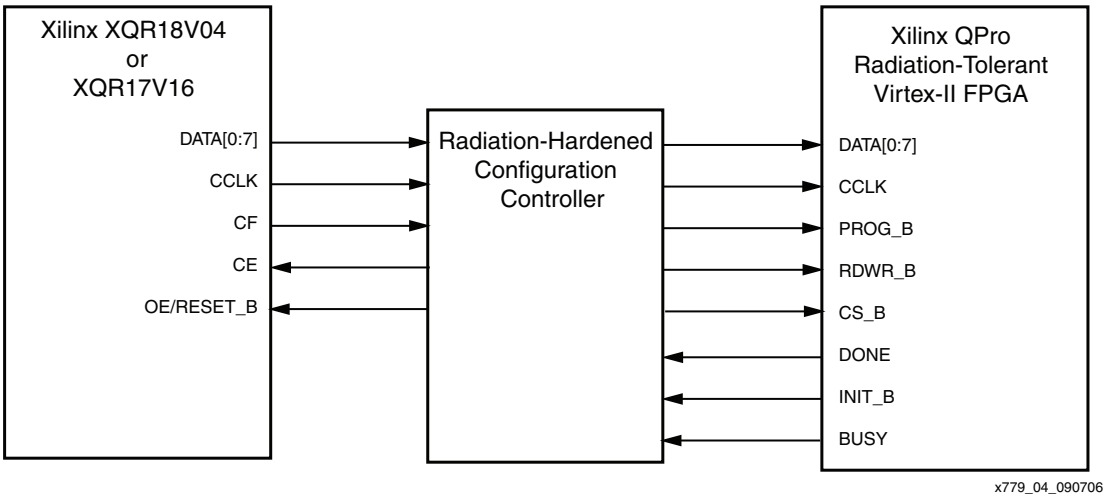


*Figure 3:* **Overview of SEE Mitigation for Virtex-II Devices**

Various implementations can be deployed to detect and correct SEUs. The flowchart in Figure 4 demonstrates a sequence to ensure prompt detection and correction of an SEU in the configuration memory or SEFI.

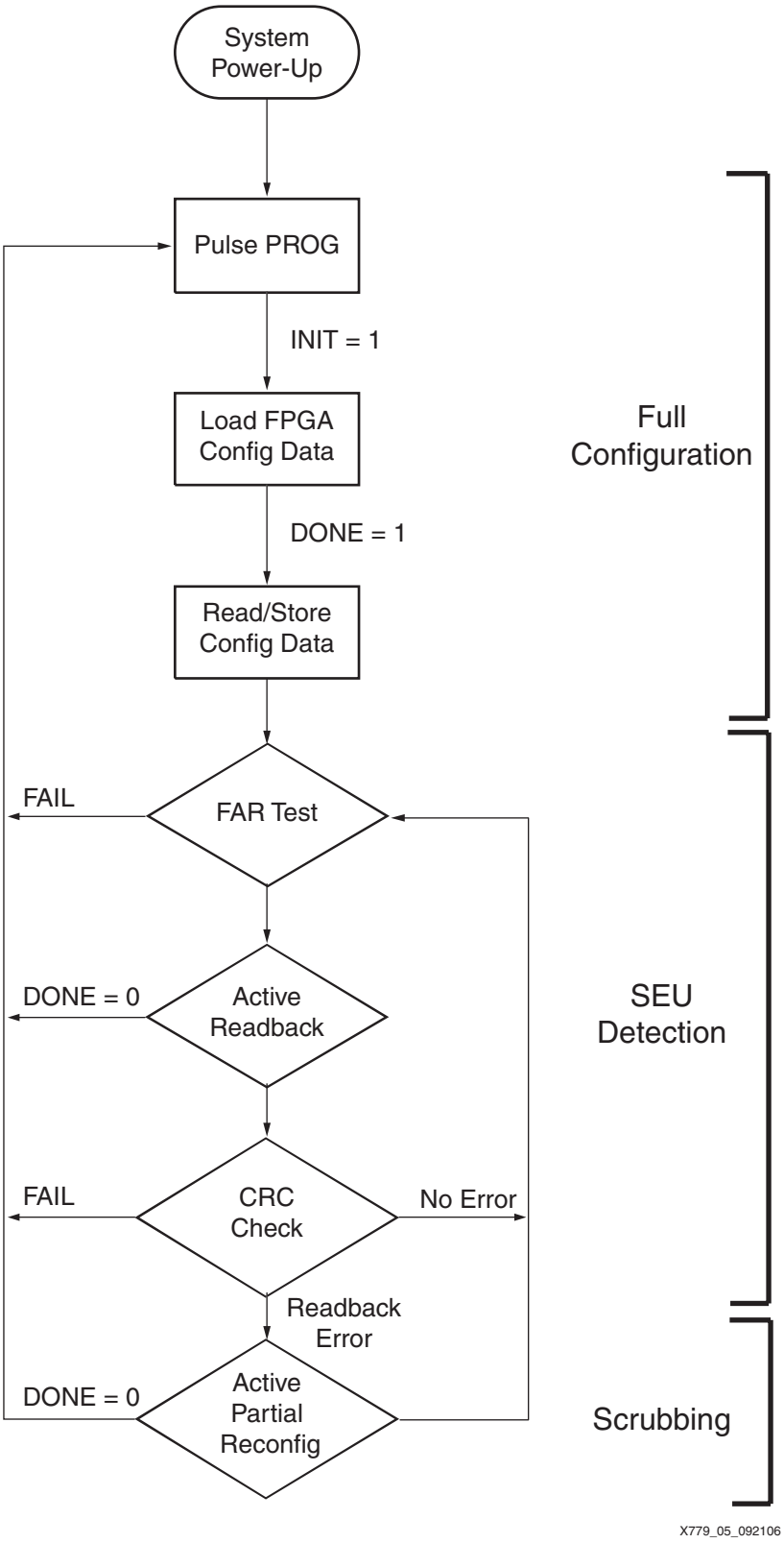


*Figure 4:* **SEU Detection and Correction Algorithm Flow Chart**

## Full Configuration

Upon power-up, the configuration management engine should pulse the PROG pin of the FPGA Low for a minimum of 300 ns. The INIT pin returning High indicates the device is ready for configuration. The configuration management engine should load the configuration from non-volatile memory, typically a PROM, into the FPGA.

Before starting a readback process to detect SEUs, two configuration registers should be readback and their values stored: configuration option register and control register. While it is also feasible to predetermine these values from the bit file, this procedure is not documented. Some important settings such as DriveDone, PWRDWN_Stat of the COR, Persist, and Security Level of the CTL could affect the SEFI detection logic; therefore, these two registers need to be scrubbed to ensure proper SEFI detection. For scrubbing purpose, a typical value for control register is `X"0000 000D"`; and a typical value for Configuration Option Register is `X"0004 3FE5"`.

*Table 10:* **Control Register**

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | | | | | | | | Reserved | SBITS | | PERSIST | Reserved | Reserved | GTS_USR_B |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | X | X | X | 0 | 0 | X |

*Table 11:* **Configuration Option Register**

| Reserved | Reserved | CRC_BYPASS | Reserved | | SHUT_RST_DCM | DONE_PIPE | DRIVE_DONE | SINGLE | OSCFSEL | | | | | | SSCLKSRC | | DONE_CYCLE | | | MATCH_CYCLE | | | LOCK_CYCLE | | | GTS_CYCLE | | | GWE_CYCLE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | X | 0 | 0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

*Table 12:* **Recommended Sequence to Readback COR or CTL Registers**

| CCLK Cycle | CS/WR | Config Data (HEX) | Explanation/Caution |
|---|---|---|---|
| 1 | CS = 0<br>WR = 1 | – | Abort Sequence to desync the FPGA. |
| 1 | CS = 0<br>WR = 0 | | |
| 8 | CS = 1<br>WR = 0 | | |
| 4 | CS = 0<br>WR = 0 | FF FF FF FF | Dummy word. |
| 4 | CS = 0<br>WR = 0 | AA 99 55 66 | Sync word. |
| 4 | CS = 0<br>WR = 0 | 30 00 80 01 | Write to CMD. |
| 4 | CS = 0<br>WR = 0 | 00 00 00 07 | Reset CRC. |
| 4 | CS = 0<br>WR= 0 | 28 00 A0 01 (CTL)<br>28 01 20 01 (COR) | Read configuration register. |
| 4 | CS = 0<br>WR= 0 | 00 00 00 00 | Dummy word. |
| 1 | CS = 1<br>WR = 0 | – | Switch from write to read mode. As soon as CS is deasserted, BUSY pin toggles High. BUSY pin must be pulled up as it is open-drain. |
| 1 | CS = 1<br>WR = 1 | | |
| 1 | CS = 0<br>WR = 1 | | |
| X+4 | CS = 0<br>WR = 1 | – | The rising CCLK edge after BUSY returns Low indicates the first valid data byte is available. |

## SEU Detection

When the FPGA is configured and ready for SEU detection, before a complete configuration memory readback is performed, a frame address register test (FAR test) should be performed to reveal possible SEFI issues. A FAR test first writes a predefined value to the FAR register followed by the configuration command to readback the FAR value. If the value matches the expected value, the configuration management engine should proceed to device readback. If the FAR readback data does not match the expected value, then a SEFI has likely occurred and the configuration engine should re-initiate a full device configuration. Upon a FAR test failure, the designer can chose to repeat the test before declaring an SEFI condition as it is possible, however unlikely, that the FAR readback was corrupted by an SET rather that an actual SEFI.

The Table 13 provides the procedure for performing the FAR TEST.

*Table 13:* **Recommended Sequence for FAR Test**

| CCLK cycle | CS/WR | Config Data (HEX) | Explanation/Caution |
|---|---|---|---|
| 1 | CS = 0<br>WR = 1 | – | Abort Sequence to desync the FPGA. |
| 1 | CS = 0<br>WR = 0 | | |
| 8 | CS = 1<br>WR = 0 | | |
| 4 | CS = 0<br>WR = 0 | FF FF FF FF | Dummy word. |
| 4 | | AA 99 55 66 | Sync word. |
| 4 | | 30 00 80 01 | Write to CMD. |
| 4 | | 00 00 00 07 | Reset CRC. |
| 4 | | 30 00 20 01 | Write to FAR. |
| 4 | | F7 4E 24 00 | Predefined FAR Value. |
| 4 | | 28 00 20 01 | Read FAR. |
| 4 | | 00 00 00 00 | Dummy Word. |
| 1 | CS = 1<br>WR = 0 | – | Switch from write to read mode. As soon as CS is deasserted, BUSY pin toggles High. BUSY pin must be pulled up as it is open-drain. |
| 1 | CS = 1<br>WR = 1 | | |
| 1 | CS = 0<br>WR = 1 | | |
| X+4 | CS = 0<br>WR = 1 | – | The rising CCLK edge after BUSY returns Low indicates the first valid data byte is available. |

Upon successful FAR test, Status Register (Table 14) should be readback and verified. If bit 5 or 6 of the status register returns 0, the device has 3-stated outputs or the inability to update synchronous device components, respectively. To mitigate this problem, the device must first be SHUTDOWN followed by a STARTUP sequence. However, since the cross section of such upset is extremely small, it is best to simply reconfiguration the device.

If bit 7 of the status register is upset, the device also loses its functionality. Reconfiguration of the device mitigates this upset.

*Table 14:* **Satus Register**

| | | | | | | | | | | | | | | | | BAD_KEY_SEQ | DEC_ERROR | ID_ERROR | DONE | INIT | MODE | | | GHIGH_IB | GWE | GTS_CFG_B | IN_ERROR | DCI_MATCH | DCM_LOCK | PART_SECURED | CRC_ERROR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

The Table 15 provides the procedure for performing the Status Register read.

*Table 15:* **Recommended Sequence for Status Register Read**

| CCLK cycle | CS/WR | Config Data (HEX) | Explanation/Caution |
|---|---|---|---|
| 1 | CS = 0<br>WR = 1 | – | Abort Sequence to desync the FPGA. |
| 1 | CS = 0<br>WR = 0 | | |
| 8 | CS = 1<br>WR = 0 | | |
| 4 | CS = 0<br>WR = 0 | FF FF FF FF | Dummy word. |
| 4 | | AA 99 55 66 | Sync word. |
| 4 | | 30 00 80 01 | Write to CMD. |
| 4 | | 00 00 00 07 | Reset CRC. |
| 4 | | 28 00 E0 01 | Read Status Register. |
| 4 | | 00 00 00 00 | Dummy Word. |
| 1 | CS = 1<br>WR = 0 | – | Switch from write to read mode. As soon as CS is deasserted, BUSY pin toggles High. BUSY pin must be pulled up as it is open-drain. |
| 1 | CS = 1<br>WR = 1 | | |
| 1 | CS = 0<br>WR = 1 | | |
| X+4 | CS = 0<br>WR = 1 | – | The rising CCLK edge after BUSY returns Low indicates the first valid data byte is available. |

After the FAR test and Status Register check have passed, a device readback should be performed. A device readback can detect general SEUs and also ensure the integrity of scrubbing. If the device does not contain any SRL16 or LUTRAM primitives, then the readback data can be directly verified against the configuration data stored in non-volatile memory.

If the FPGA design does not utilize any block memory elements (BRAM), then the configuration management engine could simply readback the entire configuration memory in one sequence. Otherwise, a complete readback requires a three-readback sequence to avoid corrupting the BRAM content or output.

The first readback sequence covers all GCLK, IOI, CLB, and part of the IOB column frames. The last IOB frame is omitted to prevent loading the first BRAM column frame into the pipeline and thus corrupting the BRAM. The last IOB frame is safely extracted in a separate sequence.

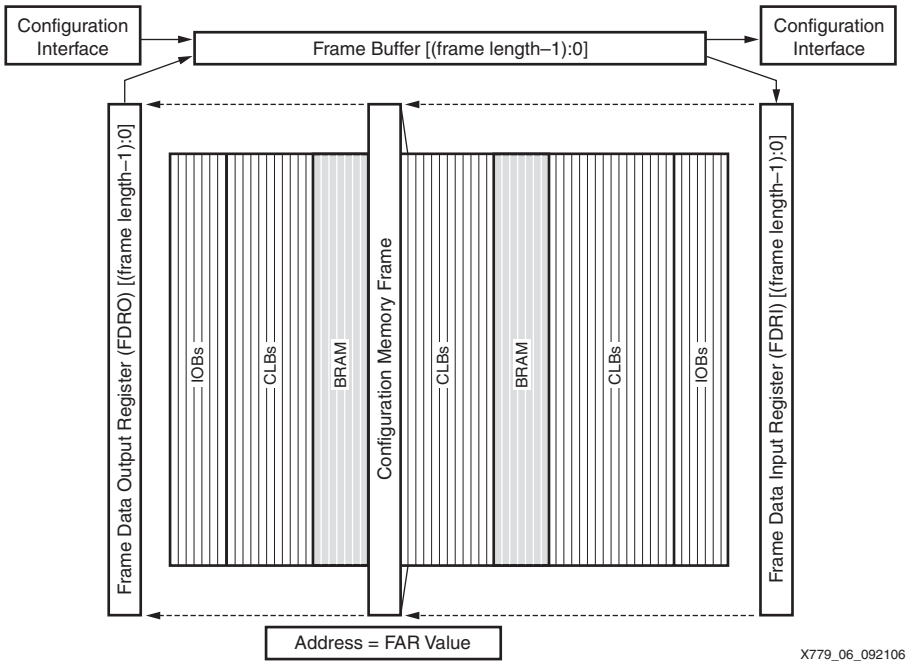


*Figure 5:* **Virtex-II Configuration Frame Data Path**

The second readback sequence reads out just the last IOB frame stored in the pipeline. The third readback sequence verifies all the rest of configuration frames controlling the BRAM interconnects. These frames should be verified even if BRAMs are not used in the design.

For second readback sequence, FAR is set to target an unused address. During readback of this unused address, the first dummy frame read back is the last IOB frame stored in the pipeline during the first readback sequence.

During an active readback, the readback data can be verified either against the configuration data stored in the non-volatile memory or compared using a CRC checksum. The CRC checksum can be on a single frame or device basis. The readback checking should be done to ensure that the previously identified SEUs are indeed corrected. A "sticking" or non-correctable SEU indicates a SEFI has resulted in an inability to write to configuration memory.

If the CRC check passes and there are no readback errors, then there are no upsets in the configuration memory. At the designer's option, the configuration management engine can either repeat the readback sequence until an upset is detected before scrubbing, or just automatically continue onto the scrubbing sequence.

If the CRC check passes but readback errors are detected, then scrubbing should follow to correct the SEUs.

When a CRC check is calculated, it should be stored for the next readback cycle for comparison purposes. If the CRC check fails, then it should be checked against the CRC calculated from the previous readback sequence. If two consecutive readbacks following scrub cycles (scrub, readback, scrub, readback) both produce the same identical incorrect CRC value (does not match the golden CRC), then this indicates that a SEFI condition has prevented successful write operations to the configuration memory. A full device reconfiguration should follow.

*Table 16:* **Readback Sequence**

| Steps | CCLK Cycle | CS/WR | Config Data (HEX) | Explanation/Caution |
|---|---|---|---|---|
| 1 | 1 | CS = 0<br>WR = 1 | – | Abort Sequence to desync the FPGA. |
| 2 | 1 | CS = 0<br>WR = 0 | | |
| 3 | 1 | CS = 1<br>WR = 0 | | |
| 4 | 4 | CS = 0<br>WR = 0 | FF FF FF FF | Dummy word. |
| 5 | 4 | | AA 99 55 66 | Sync word. |
| 6 | 4 | | 30 00 80 01 | Write to CMD. |
| 7 | 4 | | 00 00 00 07 | Reset CRC. |
| 8 | 4 | | 30 00 20 01 | Write to FAR. |
| 9 | 4 | | 00 00 00 00 or<br>01 FF FF FF or<br>04 00 00 00 | FAR for 1st readback sequence.<br>FAR for 2nd readback sequence.<br>FAR for 3rd readback sequence. |
| 10 | 4 | | 30 00 80 01 | Write to CMD. |
| 11 | 4 | | 00 00 00 04 | Read Configuration Data |
| 12 | 4 | | 28 00 60 00 | Access Frame Data output Register for read back. |
| 13 | 4 | | 48 XX XX XX | Number of readback words.(1) |
| 14 | 4 | | 00 00 00 00 | Dummy word. |
| 15 | 1 | CS = 1<br>WR = 0 | – | Switching from Write to Read Mode. |
| 16 | 1 | CS = 1<br>WR = 1 | | |
| 17 | 1 | CS = 0<br>WR = 1 | | |
| 18 | X(2) | CS = 0<br>WR = 1 | | The rising CCLK edge after BUSY returns Low indicates the first valid data byte is available. |

**Notes:**

1. The number of words is dependent on the Frame Length and the number of frames to be readback.
   Equation Example for the 1st readback on 2V1000.
   Readback Word = Frame Length × (1 dummy frame + number of frames to be readback) = 106 × {1 + 4 (all GCLK column frames) + 8 (all IOB column frames) + 43 (all but one IOI column frames) + 704 (all CLB column frame) } = 80560 = Hex 01 3A B0.
   Equation Example for the 2nd readback on 2V1000.
   Readback Word = Frame Length × (1 dummy frame + 1 IOI frame) = 106 × 2 = 212 = Hex 00 00 D4
   Equation Example for the 3rd readback on 2V1000.
   Readback Word = Frame Length × (1 dummy frame + All BRAM interconnect column frames) = 106 × (1 + 88) = 9394 = Hex 00 24 DA.
   For more information on number of frames per other devices, refer to Table 1 and Table 2.
2. The number of CCLK cycles equals to number of readback words specified in step 13 multiply by 4.

## Active Reconfiguration (Scrubbing)

After a non-SEFI SEU event is detected, the FPGA needs to be scrubbed to prevent SEU accumulation. After scrubbing completes, readback should immediately follow to ensure SEUs were corrected.

***Note:*** During scrubbing, the user design is not interrupted.

As with readback operations, scrubbing operations can potentially corrupt LUT memory primitives such as SRL16s and LUTRAM. Therefore, the design should not contain SRL16 or LUTRAM to simplify the scrubbing process. While BRAMs can be safely used, doing so requires a two-scrubbing processes: the first scrubbing sequence to cover all GCLK, IOB, IOI, and CLB columns; the second sequence to cover just the BRAM interconnect columns.

*Table 17:* **Scrubbing Sequence**

| Steps | CCLK Cycle | CS/WR | Config Data (HEX) | Explanation/Caution |
|---|---|---|---|---|
| 1 | 1 | CS = 0<br>WR = 1 | – | Abort Sequence to desync the FPGA. |
| 2 | 1 | CS = 0<br>WR = 0 | | |
| 3 | 1 | CS = 1<br>WR = 0 | | |
| 4 | 4 | CS = 0<br>WR = 0 | FF FF FF FF | Dummy word. |
| 5 | 4 | | AA 99 55 66 | Sync word. |
| 6 | 4 | | 30 00 80 01 | Write to CMD. |
| 7 | 4 | | 00 00 00 07 | Reset CRC. |
| 8 | 4 | | 30 01 C0 01 | Write to Device ID. |
| 9 | 4 | | XX XX XX XX | Device ID. Refer to User Guide for the specific device IDs. |
| 10 | 4 | | 30 01 60 01 | Write to Frame Length Register. |
| 11 | 4 | | XX XX XX XX(1) | Set Frame Length Register. Refer to User Guide for device specific frame length. |
| 12 | 4 | | 30 01 20 01 | Write to COR. |
| 13 | 4 | | XX XX XX XX(2) | COR Value. |
| 14 | 4 | | 30 00 C0 01 | Write to MASK. |
| 15 | 4 | | 00 00 00 3F | MASK Value. |
| 16 | 4 | | 30 00 A0 01 | Write to CTL. |
| 17 | 4 | | XX XX XX XX(2) | CTL Value. |
| 18 | 4 | | 30 00 C0 01 | Write to MASK. |
| 19 | 4 | | 00 00 00 00 | Zeros to disable writes to CTL. |
| 20 | 4 | | 30 00 20 01 | Write to FAR. |
| 21 | 4 | | 00 00 00 00 or<br>04 00 00 00 | FAR for 1st scrub.<br>FAR for 2nd scrub. |
| 22 | 4 | | 30 00 80 01 | Write to CMD. |
| 23 | 4 | | 00 00 00 01 | Write Configuration Data. |
| 24 | 4 | | 30 00 40 00 | Access Frame Data Input Register. |
| 25 | 4 | | 50 XX XX XX(3) | Number of words to be written. |
| 26 | X(4) | | | Config Data. |

**Notes:**

1. Configuration Frame Length starts counting from 0. Therefore, for 2V1000 which has frame length of 106 words, its frame length value should be set to 105 or hex 00 00 00 69.
2. COR and CTL value can either be obtained through readback of these registers or predetermined in the bit file. Writing to COR, MASK, and CTL can be optional for the second scrub sequence.
3. The number of word is dependent on the Frame Length and the number of frames to be readback.
   Equation Example for the 1st scrub on 2V1000.
   Config Word = Frame Length × (Config frames + 1 dummy frame) = 106 × {4 (all GCLK column frames) + 8 (all IOB column frames) + 44 (all IOI column frames) + 704 (all CLB column frame) + 1 } = 80666 = 01 3B 1A.
   Equation Example for the 2nd scrub on 2V1000.
   Config Word = Frame Length × (All BRAM interconnect Frames + 1 dummy frame) = 106 × 89 = 9434 = hex 00 24 DA.
   For more information on number of frames per other devices, refer to the Configuration Details section in the *Virtex II User Guide.*
4. Number of CCLK cycle equals to the number of Config words specified in step 25 multiple by 4.

## Other Design Considerations

### JTAG Port

Both the SelectMAP and JTAG controller logic are capable of controlling the internal configuration bus. However, the JTAG controller has overriding access over the SelectMAP logic. To further decrease the SelectMap SEFI cross section, constantly clocking 1s on the TMS of the JTAG port keeps the JTAG TAP controller in reset during readback and scrubbing. This technique prevents the JTAG interface from accidentally taking over the internal configuration bus while SelectMAP operations are active.

Scrubbing via the JTAG interface is also possible. JTAG has the advantage of less SEFI cross-section; however, it is a serial interface, requiring overhead commands such as CFG_IN and CFG_OUT to access the configuration logic. The configuration commands and concept for SelectMap mode still apply for JTAG interface as it just a different interface for accessing the configuration logic.

### Recommended Pull-Ups

BUSY is open-drain; therefore a 1 kΩ pull-up on the line is recommended. A pull-up on PWRDWN (the power-down pin) should exist to minimize SEFI. A pull-up on HSWAP_EN ensures all unused I/Os are 3-stated.

### Scrub/Readback Operation Rate

Running readback and scrub at a rate of at least 10 times faster than the expected upset rate is highly encouraged. Doing so guarantees prompt SEU detection and correction.

### Other Readback Consideration

When the golden CRC is calculated from the PROM data during FPGA programming, the golden CRC calculated should exclude BRAM contents and last frame write (LFRM). Two golden CRC can also be used, one for CLB columns and one for BRAM Interconnects.

When calculating a readback CRC for CRC SEFI detection, the first dummy frame data should be excluded since the golden CRC generated during FPGA programming does not include the dummy frame. The readback sequence can also be reduced to a two-readback sequence. The first sequence covers all CLB columns while the second sequence reads out BRAM interconnects columns including the dummy frame (the last IOB frame in the pipeline) from the BRAM_INT readback sequence in the CRC calculation.

### PROM Consideration

When power-cycling the Xilinx PROMs, its input power must reach 0V before powering back on to avoid any potential brown-out condition.

## Conclusion

Design mitigation techniques can harden applications against single events upsets. However, these upsets must be corrected so that errors do not accumulate. The configuration manager state machine example described in this application note can be adjusted to fit specific mission requirement, and when used in conjunction with SEU mitigation techniques, such as the use of TMRTool or system-level device triplication, prevents accumulating SEUs and ensures design integrity.

## Revision History

The following table shows the revision history for this document.

| Date | Version | Revision |
|---|---|---|
| 09/21/06 | 1.0 | Initial Xilinx release. |
| 02/19/07 | 1.1 | • Added details on the Status Register and the recommended procedure for reading, starting on page 12.<br>• Updated "Recommended Pull-Ups," page 18. |