Understanding Single Event Functional Interrupts in FPGA Designs

WP-01207-1.0 White Paper

This paper provides the definition for single event functional interrupts (SEFI), proposes an extension to the JEDEC definition of SEFI for programmable devices, introduces the concept of a SEFI ratio, and introduces some capability that can be used to tune the SEFI ratio for your design.

Introduction

As single event upset (SEU) mitigation becomes important for digital designs in certain markets, more design teams are focusing on using device and design techniques to ensure the majority of SEU events do not cause a system failure or interrupt, thus maximizing the uptime of the system. The following sections will explain the JEDEC definition of a functional interrupt, establish the concept of the SEFI ratio, and cover two basic categories of system design consideration. This paper also includes a discussion of programmable devices relative to the JEDEC definition where it is a combination of both the device or process and the user design that affects the SEFI ratio. Finally, this paper covers some of the basic design techniques that are used today to create a targeted SEFI ratio.

What Is SEFI and Why Is It Important?

The JEDEC definition of a SEFI is specified in the JEDEC specification JESD89A (Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices) as:

> *A soft error that causes the component to reset, lock-up, or otherwise malfunction in a detectable way, but does not require power cycling of the device (off and back on) to restore operability, unlike single-event latch-up (SEL), or result in permanent damage as in single event burnout (SEB).*

A SEFI is often associated with an upset in a control bit or register.

Let's walk through this definition in detail. First what is the definition of a soft error? Again, according to the JESD89A specification:

> *Soft errors are nondestructive functional errors induced by energetic ion strikes. Soft errors are a subset of single event effects (SEE), and include SEUs, multiple-bit upsets (MBU), SEFI, single-event transients (SET) that, if latched, become SEU, and SEL where the formation of parasitic bipolar action in CMOS wells induce a low impedance path between power and ground, producing a high current condition (SEL can also cause latent and hard errors).*

For more information and background on soft errors, refer to the *Introduction to Single Event Upsets White Paper*.

101 Innovation Drive
San Jose, CA 95134
www.altera.com

Therefore, a soft error is the upset of the state of a storage element; for example, latch, flip flop or SRAM cell, which can be fixed without power cycling the device. It is extremely rare for a semiconductor to reset or lock-up due to a soft error so the focus of this paper will be on the phrase "or otherwise malfunction in a detectable way." This paper will posit that the JEDEC definition of SEFI is correct but could be misinterpreted when it comes to programmable devices. Let's use an example of an embedded SRAM block with the ability for a user design to either use, or not use the built-in error correction code (ECC) capability. For the customer who doesn't use ECC, and an SEU changes the state of a bit in the SRAM, then when that address location is read from the SRAM – it could be possible for the device to malfunction in a detectable way. However, for a customer who uses the ECC capability, then the ECC logic will correct the bit, and the device will not malfunction overall; but the ECC logic itself will report that there has been a correction to the word. So, from a strict interpretation of the JEDEC definition – it could be claimed that the device had malfunctioned in a detectable manner – but in such a way that the overall user logic was unaffected. Therefore, in this second case, the event should not be considered a SEFI because there was no interruption to the user logic. However, in the first case there may have been a detectable user logic impact.

The Concept of SEFI Ratio

When looking at SEFI for a programmable device, both the inherent device and process capabilities need to be taken into account along with the user design. The user design's usage of both the SEU mitigation capabilities of the device as well as higher-level user design techniques for SEU mitigation, such as end-to-end cyclic redundancy check (CRC) on packets in a packet processing or networking application will dramatically affect the SEFI ratio of the design.

The industry has created the concept of a SEFI ratio, which is usually defined as the number of SEE divided by the number of functional interrupts. With this definition the ultimate goal would be to have unlimited number of SEE with no function interrupts. However, with today's deep sub-micron processes the realistic goal is to drive for a SEFI ratio that meets your system design requirements.

From an SEE standpoint there are two main types of problems that customers are trying to solve. One is for systems where incorrect information can never go outside of the system, and the other is for systems where incorrect information can go outside of the system for a short time and the goal is to minimize the time and amount of bad data. Here are two examples for the first type of problem—high-reliability servers and fast financial trading. In the first case, it is easy to see that you would not want any incorrect financial transactions to ever leave the system. In the latter case, where it is allowable for incorrect information to go outside of the system, an example is networking. If a few bad packets were to be transmitted, then depending on the direction of traffic either the upstream or downstream systems would detect that these are bad packets and request retransmissions. So while not desirable, it is not catastrophic for there to be some retransmissions of packets in a network.

These two cases drive different but related requirements for semiconductor devices. For the first case, the key criteria is the time it takes to detect an SEE. For the second case, the time it takes to detect is equally important as the time to correct an SEE. A fast detect time coupled with a fast correction time to get the system back to a normal operation is desired.

SEFI Ratio Tuning

Additionally some customers have definitions for different levels of severity for an SEU. This approach is supported by using Altera's hierarchical tagging in conjunction with sensitivity processing. Using these capabilities enables a user design to compare the location of the SEU event to the criticality as defined for their design. Using this type of approach, the focus is on reducing the SEFI ratio for critical parts of the design while other parts of the design can experience SEU events with no impact to the SEFI ratio. For example, triple modular redundancy (TMR) can be used for key state machines so that SEU events to one part of the triplicated state machine will not cause a functional interrupt to the design.

During the design phase, fault injection is used to characterize the design for SEFI robustness; enabling designers to ensure they have reached the necessary SEFI ratio or failures in time (FIT) rate for the design. Often suppliers have multiple types of fault injection to support different needs of the design teams. Altera has both the fault injection register as well as the partial reconfiguration-based fault injection. The fault injection register allows an easy, fast way to test the overall system design response to a SEU event by modifying the appropriate bit(s) going into the error detection cyclic redundancy check (EDCRC). The fault injection based on leveraging the partial reconfiguration capability inside the device injects changes into the CRAM bit(s) themselves, which enables SEFI characterization.

For additional details about Altera's SEU mitigation capabilities, refer to the *Enhancing Robust SEU Mitigation with 28 nm FPGAs White Paper*.

Conclusion

For your FPGA designs, where you have a FIT rate target, take advantage of the SEU mitigation capabilities available in the device. Then include additional mitigation capabilities in the design as needed. Finally test the design with fault injection and use those results to further improve the SEFI ratio for your design.

Further Information

- Single event upsets web page: www.altera.com/support/devices/reliability/seu/seu-index.html
- White Paper: *Introduction to Single Event Upsets* www.altera.com/literature/wp/wp-01206-introduction-single-event-upsets.pdf
- White Paper: *Enhancing Robust SEU Mitigation with 28 nm FPGAs* www.altera.com/literature/wp/wp-01135-stxv-seu-mitigation.pdf

Acknowledgements

- Michael Sydow, Sr. Product Marketing Manager, High-End Products, Altera Corporation

Document Revision History

Table 1 lists the revision history for this document.

Table 1. Document Revision History

Date	Version	Changes
September 2013	1.0	Initial release.