Intel® Agilex™ SEU Mitigation User Guide

Updated for Intel® Quartus® Prime Design Suite: 19.3
1. Intel® Agilex™ SEU Mitigation Overview.................................................................................................................................................................3
   1.1. SEU Mitigation Techniques..........................................................................................................................3
   1.2. Configuration RAM....................................................................................................................................4
   1.3. Memory Blocks........................................................................................................................................4
   1.4. Triple Modular Redundancy.......................................................................................................................5
   1.5. Failure Rates.............................................................................................................................................5
2. Intel Agilex CRAM Error Mitigation..........................................................................................................................................................7
   2.1. Error Message Queue..............................................................................................................................7
   2.2. Scrubbing..................................................................................................................................................8
3. Intel Agilex SEU Mitigation Implementation Guides.................................................................................................................................10
   3.1. Setting the SEU_ERROR Pin.....................................................................................................................10
   3.2. Setting up the SEU Mitigation Features..................................................................................................10
   3.3. Enabling Priority Scrubbing.....................................................................................................................11
1. Intel® Agilex™ SEU Mitigation Overview

Single event upsets (SEUs) are rare and unintended changes in the internal memory elements of an FPGA caused by cosmic radiation. The memory state change is a soft error with no permanent damage but the FPGA may operate erroneously until background scrubbing fixes the upset.

Because of the low chance of occurrence, your design may not require SEU mitigation. However, if your system includes multiple FPGAs and requires very high reliability and availability, consider using mitigation techniques to detect and recover from SEU errors.

Related Information
• Introduction to Single-Event Upsets
• Understanding Single Event Functional Interrupts in FPGA Designs

1.1. SEU Mitigation Techniques

The Intel® Quartus® Prime software offers several features to detect, correct, and characterize the effects of SEU on your designs. Additionally, Intel Agilex™ FPGAs contain dedicated circuitry to help detect and correct errors.

Intel Agilex SEU mitigation features can benefit the system by:
• Ensuring the system functions properly at all time
• Preventing a system malfunction caused by an SEU event
• Handling the SEU event if it is critical to the system

Table 1. SEU Mitigation Areas and Approaches for Intel Agilex Devices

<table>
<thead>
<tr>
<th>Area</th>
<th>SEU Mitigation Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error detection and correction</td>
<td>Enable the error detection and correction (EDC) feature to detect CRAM SEU events and automatically correct the CRAM contents.</td>
</tr>
<tr>
<td>Memory block error correction code</td>
<td>Take advantage of the error correction code (ECC) feature and the special layout design of the Intel Agilex M20K memory blocks to reduce SEU failures in time (FIT) rate to almost zero.</td>
</tr>
<tr>
<td>SEU sensitivity processing(1)</td>
<td>Use the sensitivity processing feature to identify if the SEU on a CRAM bit location is critical to the function of your compiled FPGA design bitstream file.</td>
</tr>
</tbody>
</table>

(1) The feature will be available in a future Intel Quartus Prime release.
### Area | SEU Mitigation Approach
--- | ---
Fault injection(1) | Use the fault injection feature to help you validate system response to the SEU event by intentionally changing the CRAM state to trigger an error.
Hierarchy tagging(1) | Use hierarchy tagging, together with sensitivity processing and fault injection, to report SEU and constrain error injection to specific portions of your design logic.
Triple modular redundancy | Use triple modular redundancy (TMR) technique on critical logic such as state machines to improve hardware fault tolerance.

### 1.2. Configuration RAM

FPGAs use memory in user logic (bulk memory and registers) and in configuration RAM (CRAM). The Intel Quartus Prime Programmer loads the CRAM with your design (.sof file). During device configuration, the CRAM configures all FPGA logic and routing.

If an SEU strikes a CRAM bit that is not in use, the effect can be harmless. However, if the affected CRAM bit is in use for critical internal signal routing or lookup table logic bits, the device may experience a functional error.

**Related Information**

Intel Agilex Configuration User Guide

Provides more information about CRAM and user design in Intel Agilex devices.

### 1.3. Memory Blocks

Intel Agilex devices contain three types of memory blocks: Embedded SRAM (eSRAM) blocks, M20K blocks, and memory logic array blocks (MLABs). The M20K blocks and eSRAM blocks support ECC. The ECC feature detects and corrects data errors at the output of the memory.

**Note:** When you engage the ECC feature, you cannot use the byte enable and coherent read features.

**Table 2. ECC for M20K and eSRAM Blocks**

<table>
<thead>
<tr>
<th>Item</th>
<th>M20K Block</th>
<th>eSRAM Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Built-in support</td>
<td>In ×32-wide simple dual-port mode</td>
<td>In ×64-wide simple dual-port mode.</td>
</tr>
<tr>
<td>Features</td>
<td>32-bit word error detection and correction:</td>
<td>64-bit word error detection or correction:</td>
</tr>
<tr>
<td></td>
<td>• Single-error correction</td>
<td>• Single-error correction</td>
</tr>
<tr>
<td></td>
<td>• Double-adjacent-error correction</td>
<td>• Double-error detection</td>
</tr>
<tr>
<td></td>
<td>• Triple-adjacent-error correction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The ECC cannot guarantee detection or</td>
<td></td>
</tr>
<tr>
<td></td>
<td>correction of non-adjacent two-bit (or more)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>errors.</td>
<td>errors.</td>
</tr>
<tr>
<td>Flags indicating memory status</td>
<td>• e—error</td>
<td>• c(7:0)_error_correct_0—error corrected</td>
</tr>
<tr>
<td></td>
<td>• ue—uncorrectable error</td>
<td>• c(7:0)_error_detect_0—error detected</td>
</tr>
<tr>
<td></td>
<td>The status flags are part of the regular</td>
<td></td>
</tr>
<tr>
<td></td>
<td>outputs from the memory block.</td>
<td></td>
</tr>
</tbody>
</table>
When you engage ECC, the M20K memory runs slower than in non-ECC simple dual-port mode. To achieve a higher performance—compared to non-pipeline ECC mode—at the expense of a one-cycle latency, enable the optional ECC pipeline registers before the output decoder.

**Related Information**
  Provides more information about implementing ECC with the embedded memory Intel FPGA IP cores.
- Memory Blocks Error Correction Code Support, Intel Agilex Embedded Memory User Guide
  Provides more information about ECC in Intel Agilex memory blocks.

### 1.4. Triple Modular Redundancy

Triple modular redundancy (TMR) is an established SEU mitigation technique for improving hardware fault tolerance. Use TMR if your system cannot suffer downtime caused by an SEU.

A TMR design has three identical instances of hardware with a voting hardware at the output. If an SEU affects one of the hardware instances, the voting logic notes the majority output. This operation masks malfunctioning hardware.

With TMR, your design does not suffer downtime in the case of a single SEU:
- When the system detects a faulty module, the system scrubs the error by reprogramming the module.
- The error detection and correction time is many orders of magnitude less than the mean time between failures (MTBF) of SEU events.
- The system can repair a soft interrupt before another SEU affects another instance in the TMR application.

The disadvantage of TMR is that, in addition to voting logic, it requires three times more hardware cost than a non-TMR design. To minimize the hardware cost, implement TMR for only the most critical parts of your design.

You can automate generation of TMR designs by automatically replicating designated functions and synthesizing the required voting logic. For example, Synopsys® offers a tool that automate TMR synthesis.

**Related Information**
High-Reliability Design: No Room for Error, Synopsys website

### 1.5. Failure Rates

The soft error rate (SER) or SEU reliability is expressed in Failures In Time (FIT)—the number of failures you can expect in one billion operation hours.

For example, a design with 5,000 FIT experiences a mean of 5,000 SEU events in $10^9$ hours (114,155.25 years). Because SEU events are statistically independent, FIT is additive. If a single FPGA has 5,000 FIT, then ten FPGAs have 50,000 FIT (50,000 failures in 114,155.25 years).
Another reliability measurement is the mean time to failure (MTTF), which is the reciprocal of the FIT or 1/FIT. For a FIT of 5,000 in standard units of failures per billion hours, MTTF is $1/(5,000/1Bh) = 1 \text{ billion}/5,000 = 200,000 \text{ hours} = 22.83 \text{ years}$.

SEU events follow a Poisson distribution, and the cumulative distribution function (CDF) for mean time between failures (MTBF) is an exponential distribution. For more information about failure rate calculation, refer to the *Intel FPGA Reliability Report* (available upon request).

Neutron SEU incidence varies by altitude, latitude, and other environmental factors. The Intel Quartus Prime software provides SEU FIT reports based on compiles for sea level in Manhattan, New York. The JESD89A specification defines the test parameters.

**Tip:** You can convert the data to other locations and altitudes using calculators, such as the one at seutest.com. You can adjust the SEU rates in your project by including the relative neutron flux in your project's `.qsf` file.

**Related Information**

- Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray Induced Soft Errors in Semiconductor Devices (JESD89A), JEDEC* website
- Flux Calculation, seutest.com
2. Intel Agilex CRAM Error Mitigation

Intel Agilex devices feature on-chip EDC circuitry to detect soft errors. If you enable the internal scrubbing feature, the Intel Agilex FPGA corrects an error caused by an SEU event if it is correctable.

Table 3. Detection and Correction of Error Types

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Detection</th>
<th>Correction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single bit error</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Double adjacent errors</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>Multiple bit errors</td>
<td>The feature will be available in a future Intel Quartus Prime release.</td>
<td>—</td>
</tr>
</tbody>
</table>

Note: For information about the embedded memory ECC feature, refer to the related information.

Related Information

  Provides more information about implementing ECC with the embedded memory Intel FPGA IP cores.
- Memory Blocks Error Correction Code Support, Intel Agilex Embedded Memory User Guide
  Provides more information about ECC in Intel Agilex memory blocks.

2.1. Error Message Queue

When it detects an SEU error, the Intel Agilex device stores the error information in the error message queue. The queue can store up to four different messages. Each error message records the sector address, type, and location of the error.

The `SEU_ERROR` signal goes high whenever the error message queue contains one or more error messages. The signal stays high if there is an error message in the queue. The `SEU_ERROR` signal goes low only when the SEU error message queue is empty—after you shift out all the error messages. You must set the `SEU_ERROR` pin function to observe the `SEU_ERROR` pin behavior.

To retrieve the error message queue contents, use these tools:

- Intel Quartus Prime Fault Injection Debugger
- Advanced SEU Detection Intel FPGA IP

The feature will be available in a future Intel Quartus Prime release.
### Table 4. Error Message Queue Bit Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Width</th>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sector address</td>
<td>32</td>
<td>31:24</td>
<td>Reserved</td>
</tr>
<tr>
<td>(Most significant 32-bit word</td>
<td></td>
<td></td>
<td>in  seu_avst_data signal)</td>
</tr>
<tr>
<td>23:16</td>
<td></td>
<td></td>
<td>Sector address of the error</td>
</tr>
<tr>
<td>15:4</td>
<td></td>
<td></td>
<td>Reserved</td>
</tr>
<tr>
<td>3:0</td>
<td></td>
<td></td>
<td>Number of errors detected in the sector-1</td>
</tr>
<tr>
<td>Error location(3)</td>
<td>32</td>
<td>31:29</td>
<td>Error type:</td>
</tr>
<tr>
<td>(Least significant 32-bit word</td>
<td></td>
<td></td>
<td>• 01—single bit</td>
</tr>
<tr>
<td>in seu_avst_data signal)</td>
<td></td>
<td></td>
<td>• 10—multi-bit</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td>Correction Status:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 0—not corrected</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• 1—corrected</td>
</tr>
<tr>
<td>27:24</td>
<td></td>
<td></td>
<td>Reserved</td>
</tr>
<tr>
<td>23:12</td>
<td></td>
<td></td>
<td>Bit position within the frame</td>
</tr>
<tr>
<td>11:0</td>
<td></td>
<td></td>
<td>Combination of row and frame index</td>
</tr>
</tbody>
</table>

### Related Information
- Setting up the SEU Mitigation Features on page 10
- Setting the SEU_ERROR Pin on page 10

#### 2.2. Scrubbing

Intel Agilex devices support automatic CRAM error correction without reloading the original CRAM contents from an external copy of the original programming bit-stream.

You can also choose to perform scrubbing using partial reconfiguration by reloading the impacted sector. Although scrubbing corrects the SEU error, the SEU error message queue keeps the SEU error message until you retrieve it.

**Internal Scrubbing**

The internal scrubbing feature automatically corrects single-bit errors.

Intel recommends that you turn on internal scrubbing. If you do not enable internal scrubbing, the device turns off the SEU mitigation feature for a sector after an error occurs in the sector. Subsequently, the device stops detection of correctable or uncorrectable SEU occurrence in the affected sector.

If you enable the internal scrubbing feature, you must still plan your recovery sequence. Although the scrubbing feature can restore the CRAM array to the intended configuration, a latency period exists between detection and correction of the soft error. During this latency period, the Intel Agilex device may be operating with errors.

---

(3) For single bit error with internal scrubbing, the error location provides the error bit position. For multiple bit errors or single bit error without internal scrubbing, bit [23:0] returns 0.
Priority Scrubbing

You can specify portions of the design as high priority sectors for internal scrubbing. The Intel Agilex EDC circuitry detects and corrects errors that occur in the high priority sectors first before detecting and correcting errors in other sectors.

Related Information

- Setting up the SEU Mitigation Features on page 10
- Enabling Priority Scrubbing on page 11
3. Intel Agilex SEU Mitigation Implementation Guides

3.1. Setting the SEU_ERROR Pin

Perform these steps to set the SEU_ERROR pin function in the Intel Quartus Prime software.

1. From the Intel Quartus Prime menu, select Assignments ➤ Device.
2. In the Device window, click Device and Pin Options.
3. In the Device and Pin Options windows, navigate to Configuration and click Configuration Pins Options.
4. In the Configuration PIN window, turn on USE SEU_ERROR output and select any unused SDM pin to implement the SEU_ERROR pin function.
5. Click OK.

Related Information
Error Message Queue on page 7
Provides information about the SEU_ERROR pin behavior.

3.2. Setting up the SEU Mitigation Features

Perform these steps to set up the Intel Agilex SEU detection, enable internal scrubbing, enable sensitivity map generation, and allowing fault injection.

1. From the Intel Quartus Prime menu, select Assignments ➤ Device.
2. In the Device window, click Device and Pin Options.
3. In the Device and Pin Options windows, navigate to Error Detection CRC.
4. Specify the following options:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable error detection check</td>
<td>Turn on to enable the error detection feature. This option is required for sensitivity processing and fault injection, or if you want to observe the SEU_ERROR pin behavior.</td>
</tr>
<tr>
<td>Minimum SEU interval</td>
<td>Enter a value of 0 to 10000 milliseconds to set the minimum time between two checks of the same bit. To check as frequently as possible, enter 0.</td>
</tr>
<tr>
<td>Enable internal scrubbing</td>
<td>Turn on to enable the error correction feature. This option is required for sensitivity processing.</td>
</tr>
</tbody>
</table>

5. Click OK.

Related Information
- Scrubbing on page 8
- Error Message Queue on page 7
  Provides information about the SEU_ERROR pin behavior.
3.3. Enabling Priority Scrubbing

To specify areas for high priority internal scrubbing, use the Intel Quartus Prime Logic Lock region and design partition features.

1. From the Intel Quartus Prime menu, select Assignments ➤ Logic Lock Regions Window.
2. In the Logic Lock Regions Window, create a region and place it within a design partition.
3. Add your critical design modules, entities, or group of logic to preserve and lock them to the region.
4. From the Intel Quartus Prime menu, select Assignments ➤ Assignment Editor.
5. In the Assignment Editor window, assign Priority SEU Area to the design partition where you place the Logic Lock region.

Alternatively, you can include the following instruction in the project's Quartus settings file (.qsf):

```
set_instance_assignment -name PRIORITY_SEU_AREA ON -to <partition name>
```

The Intel Quartus Prime software sets the internal scrubbing schedule of the priority sectors to "as fast as possible". The internal scrubbing schedule for other sectors follows the project's Minimum SEU interval global assignment.

Related Information

- Scrubbing on page 8

  Provides more information about creating and using Logic Lock regions.

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Intel Quartus Prime Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019.10.17</td>
<td>19.3</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>