Flash-Based FPGA
NEPP FY12 Summary Report

Douglas Sheldon
Jet Propulsion Laboratory
Pasadena, California
Flash-Based FPGA
NEPP FY12 Summary Report

NASA Electronic Parts and Packaging (NEPP) Program
Office of Safety and Mission Assurance

Douglas Sheldon
Jet Propulsion Laboratory
Pasadena, California

NASA WBS: 724297.40.43
JPL Project Number: 103982
Task Number: 03.02.01

Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

http://nepp.nasa.gov
This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the National Aeronautics and Space Administration Electronic Parts and Packaging (NEPP) Program.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

# TABLE OF CONTENTS

1.0 Introduction ................................................................................................................................. 1

2.0 Microsemi Flash FPGA ..................................................................................................................... 2
   2.1 Flash FPGA Technology and Fabric ............................................................................................ 2

3.0 Concerns for use in Spacecraft Applications .................................................................................. 5
   3.1 Propagation Delay in Flash FPGAs ............................................................................................. 6
   3.2 Wear Leveling FPGA Resources ................................................................................................. 8
   3.3 Radiation Related Effects .......................................................................................................... 9

4.0 Summary ....................................................................................................................................... 11

5.0 References ..................................................................................................................................... 12
1.0 INTRODUCTION

This report discusses the flash-based Field Programmable Gate Arrays (FPGA) product and technology offered by Microsemi. Flash-based FPGAs are FPGAs that use nonvolatile flash memory cells as their core building block for both interconnect routing and logical tile architecture. Flash-based FPGAs offer a unique combination of reprogramability, like SRAM-based FPGAs, while at the same time being able to retain their configuration with power removed due to the nonvolatile nature of the flash devices.

This FY12 report is a followup report based on the work accomplished in FY11 on the same flash-based FPGAs. In that FY11 report the basic flash FPGA technology and its implementation in the FPGA fabric was reviewed. The FY11 report also discussed the qualification data that Microsemi has provided. The additional work performed in FY12 focused on the sensitivity of the flash FPGA fabric to possible errors due to degradation or operation of the flash cells that make up the core logic elements. It is these types of both hard and soft errors that will limit the usefulness of this device technology in space applications.

Modern CMOS devices made in high volume commercial foundries like the Microsemi flash FPGAs are made on processes that have well-established part per million (ppm) defect levels. These types of defect levels imply that the parts will reliably operate for several years, according to their data sheet, at least over commercial temperature range, and perhaps higher if so screened.

The main reliability concern for use of these devices to NASA is second order failure mechanisms that may manifest themselves as a result of the statistical variation of the large number of transistors (much larger than ten million) that are used on each die. Statistical variation in transistors means distributions of drive current, input capacitance, and resistance variations, for example. These types of physical distributions are then coupled with complex device operation scenarios that occur with such highly integrated devices like the flash FPGA.

The end results of this combination are the possibility of sophisticated timing-related and timing-sensitive failures that often do not occur during normal qualification and screening procedures. It is these timing-related failures that are examined and discussed in this FY12 report. Timing-related failures can either be associated with radiation-induced defects, but also can be the result of operation of the high voltage circuitry needed to program and erase the flash devices, most specifically disturbance-induced failures. Integrating these operationally sensitive timing failures into the overall FPGA device qualification for use in space applications is the final recommendation of this paper.
2.0 MICROSEMI FLASH FPGA

The technology of the Microsemi flash FPGA has already been discussed at length in the FY11 report. The details of the fabric and its physical and logical implementations are reviewed here as a basis for further discussion. There are two radiation-tolerant, flash-based FPGAs offered by Microsemi for use in spacecraft operations, the RT3PE600L and the RT3PE3000L. The devices range from 300K to 3M system gates, respectively. The parts can sustain up to 700Mbps DDR I/Os and have mixed voltages between 1.2 or 1.5V at the core to 1.8/2.5/3.3V at the periphery and I/Os. This allows for 250MHz to 350MHz system operation.

2.1 Flash FPGA Technology and Fabric

The radiation-tolerant flash FPGA from Microsemi is based on the concept of the flash device as a switch, not just a memory element. At the heart of Microsemi’s flash FPGA is a custom two-cell flash transistor design [1]. This fundamental building block has two non-volatile flash transistors sharing a common floating gate. This design, shown in Figure 2.1-1, has two flash transistors, one called “sense” and the other “switching.” The sense device is used to access the cell in terms of programming and reading, and a switching device is used to set the FPGA configuration and route logic signals. The switch device is, in fact, the “switch” in the FPGA fabric. By having separate switch and sense diffusions, the switch source and drain are preserved from high programming biases and can interface directly with low-voltage logic. Each cell is used directly in the signal path. Figure 2.1-1 shows the electric schematic of the sense-switch cell.

![Figure 2.1-1. Two flash transistor core cell schematic.](image)

Having the high power regions separate from the low power regions is a key part of the FPGA fabric and layout. Each logic tile is a combination of low-voltage (LV) CMOS logic and flash switches. The core logic tiles are called VersaTiles. Each VersaTile can be configured as a three-input logic function, a D-flip-flop, or a latch by programming the appropriate flash switch interconnections. This is shown in Figure 2.1-2.

![Figure 2.1-2. VersaTile configuration options.](image)
VersaTiles can be connected with any of the four different levels of routing hierarchy. Flash-based switches are distributed throughout the FPGA to provide nonvolatile, reconfigurable interconnect programming. The individual VersaTile is shown in Figure 2.1-3.

The routing structure is designed to provide high performance through the four-level hierarchy of routing resources. This hierarchy is defined as:

1. Global signal network resources
2. High speed bus lines
3. Efficient long line resources
4. Ultrafast local lines and resources.

This is shown in Figure 2.1-4. The ultra-fast local resources are dedicated lines that allow the output of each VersaTile to connect directly to every input of the eight surrounding VersaTiles. The efficient long-line resources provide routing for longer distances and higher fan-out connections. These resources vary in length (spanning one, two, or four VersaTiles), run both vertically and horizontally, and cover the entire device.

Each VersaTile can drive signals onto the efficient long-line resources, which can access every input of every VersaTile. Routing software automatically inserts active buffers to limit loading effects. The high-speed, very-long-line resources, which span the entire device with minimal delay, are used to route very long or high fan-out nets. The lengths can be ±12 VersaTiles in the vertical direction and ±16 in the horizontal direction from a given core VersaTile.

Very long lines in low power flash devices have been enhanced over those in previous ProASIC families. This is designed to provide a performance boost for long-reach signals. The global networks are intended to be low skew, high-fanout nets that are accessible from external pins or internal logic. These nets are typically used to distribute clocks, resets, and other high fan-out nets requiring minimum skew. The VersaNet networks are implemented as clock trees, and signals can be introduced at any junction. These can be employed hierarchically, with signals accessing every input of every VersaTile.
Six global and three quadrant networks can reach each VersaTile in the die. These networks can be aggregated to map local clocks or resets or any high fan-out net. The embedded MUX tree offers a choice at the local level between sourcing from the global network or any local driver.

The global network distribution is shown in Figure 2.1-5.
3.0 CONCERNS FOR USE IN SPACECRAFT APPLICATIONS

The fundamental use of the flash cell as an enabling technology for the Microsemi ProASIC series FPGAs is also the source for concern for its high reliability spacecraft missions. Flash cells have inherent reliability and radiation issues that ultimately limit the capability and applicability of these FPGAs for NASA use.

The ultimate physics of failure issue for both reliability and radiation performance is the stability of the stored charge on the floating gate. The stability of the stored charge is a function of the operating temperature, number of times the charge has been erased and programmed on to the floating gate, and the electric fields that might disturb the charge. An example of the stability concerns of the flash cell is shown in Figure 3.0-1.

Figure 3.0-1 is a graph of the expected lifetime for the ProASIC flash cell to experience a leakage failure during a verify process as a function of temperature [1]. This type of data retention-based failure shows a strong temperature dependence with a 100-year lifetime predicted at 70°C, while only a two-year lifetime predicted at 150°C. Because of the temperature extremes that a spacecraft may experience, both planned and unexpected, this temperature-sensitive data retention remains a concern to final technology implementation that must be actively managed at all times.

![Flash Cell Lifetime vs Temperature](image)

**Figure 3.0-1.** Flash cell leakage failure lifetime versus temperature. [1]

Some flash-based memory devices have been used for many years in limited use space applications however. The key difference between these standard memories and the ProASIC line of flash-based FPGAs is ability to implement redundant flash cells. Redundant memory cells are used in all flash-based memory as the only practical means possible to provide the end user with a manageable level of bit errors. Redundant flash cells for the basis for ECC circuitry and provide the user with the ability to map data in and out of heavily used blocks and locations.

Flash-based FPGAs have no redundant elements in the architecture however. As described above, the flash cells are used throughout both the logic cell definition in the VersaTile as well as in the entire routing and connection fabric. This means the ProASIC FPGA is sensitive to the loss (or degradation) of
a single flash cell. There is no means to “switch in” redundant elements to replace cells that have been damaged or degraded. Statistical analysis of ProASIC FPGA performance is then required to adequately define the risk and performance for use in a high reliability space environment. Statistical analysis is used in this instance to reflect not only large numbers of individual parts to help understand the variation in the overall flash cell technology, but statistical analysis of the ProASIC resources and possible statistically related failure mechanisms.

Microsemi has attempted to address the statistical nature of the flash technology by providing measured flash device performance before and after 1,000-hour life test conditions. This data is shown in Figure 3.0-2 [2]. Voltage margin is defined as the difference between the programmed and erased state of the flash device. The concept of margin is to define the on or off state and how robust that difference is. Figure 3.0-2 shows the entire population of programmed and erased flash devices on several ProASIC FPGAs. This data is available by placing the FPGA in a test mode.

![Figure 3.0-2. Distribution of V_margin pre and post life stress. [2]](image)

Three separate distributions are clearly visible. The vast majority of the bits have a V_margin between 4V to 6V or between -2V to -4V. This means the final margin value is between 6 and 10V. This large amount of margin voltage allows the FPGA to continue to function while the cells shift and degrade as a function of reliability and radiation variables. These environmentally-based shifts and degradations may only amount to 10’s to 100’s of millivolts, for example. This means that the chance of a cell “flipping” from 1 to 0 or 0 to 1 is very small.

The small number of bits (<100) that have a have margin that is only 2 to 3 volts is much more of concern and hence the focus of being a “weak link” or “first fail.” While a possible 100’s of millivolts shift due to degradation may have still have a limited probability of completely flipping a cell, such a shift could have measurable impact on propagation delay.

### 3.1 Propagation Delay in Flash FPGAs

Propagation delay in flash FPGAs has become an important topic in recent engineering literature. The results are presented here in review.

Electrical simulations models of the VersaTile logic cell describe in Figure 2.1-3 above have been completed [3]. A SPICE model was made using 130nm CMOS transistor models. The transistors in the
model were sized to obtain a propagation delay similar to the component models published by Microsemi in the Designer Tool. As an example, the VersaTile is expected to have a 630ps average delay when configured as a two-input NAND gate. This delay drops to 540ps when the VersaTile is configured as an inverter. The flash floating gate switches discussed in Figure 2.1-1 were modeled as NMOS pass transistors. There are 32 flash switches in each VersaTile. This means there are several ways to map the same logic function in the available resources. Two different NAND devices can be defined, for example, NAND1 and NAND2. Degradation due to environmental effects of radiation and temperature and operational degradation due to program/erase cycles and disturb fields can all be modeled by including a current source in parallel with each NMOS pass gate, as shown in Figure 3.1-1.

The addition of the current source allows for a variety of different degradation scenarios to be defined. Current source leakage can be used to model radiation-induced damage that is the result of accumulated charge in isolation oxides, while the threshold voltages of the NMOS pass devices can be changed to represent degradation due to charge trapping from long-cycle-count program and erase conditions, for example. The results of this type of modeling the VersaTile are shown in Figure 3.1-2.

In Figure 3.1-2 two different combinations flash cells in a VersaTile cell of NAND, NOR, and INV gates are defined. This total of six different combinations is then modeled with three different degradation conditions. The three different degradation cases reflect different combinations of threshold shifts and leakage currents.

![Figure 3.1-1. NMOS pass transistor with current source for degradation.](image1)

![Figure 3.1-2. Propagation Delay in VersaTile flash FPGA cells.](image2)
Figure 3.1-2 shows that moderate leakage and degradation can result in 1–10% propagation delay. This result is somewhat independent of logical implementation. As mentioned above, this amount of delay would translate into a delay of approximately 50–60ps. As the leakage current changes and the amount of shift in the flash device becomes approximately 150mV, then the amount of propagation delay increases by more than an order of magnitude to approximately 200%. This amount of threshold voltage shift can easily be obtained with a radiation dose of approximately 10krad, or 10,000 program/erase cycles. Kastensmidt has also shown that propagation delay is a nonlinear function of accumulated dose, functionally similar to threshold degradation, as a function of program/erase cycles:

\[ V_t(dose, \text{cycles}) = V_t(\text{final}) + [V_t(\text{initial}) - V_t(\text{final})]e^{-A(dose,\text{cycles})} \]  \hspace{1cm} (1)

The existence of a nonlinear degradation effect means that it is difficult to accurately define a large safety margin for reliable use in high-reliability, low-risk tolerance space missions without severely restricting the performance window of the FPGA. This point is countered with the discussion of how much of a propagation delay can be tolerated by a circuit/design before it’s considered to cause a failure. The answer to this question always remains uniquely dependent on the design. There are rules of thumb that suggest that higher speeds, higher bandwidth designs will naturally be less tolerant of a given amount of propagation delay when compared to lower speed, lower bandwidth designs.

Managing propagation delay is part of the overall timing closure issues that are at the very heart of FPGA design and synthesis. By the very nature of FPGA architecture, modern digital designs delay can be as much as 90% due to routing and only 10% due to the core logic usage. The place and routing procedure during FPGA development will impose various constraints designed to minimize delay, but it can never be eliminated. For example, in an effort to reduce delay, many synthesis tools automatically insert buffers to provide more current drive for the capacitive loads associated with some nets. Understanding the distribution of propagation delay that is initial present in any FPGA is encompassed and addressed by the software resources provided by the FPGA vendor. Understanding how these distributions of delay may change as a result of environmental degradation (temperature, voltage, radiation) is becoming the most practical concern for modern, high reliability systems that implement high density and I/O count FPGAs. Several authors have proposed various mechanisms for addressing possible shifts and variations in FPGA timing issues.

### 3.2 Wear Leveling FPGA Resources

Wear leveling has been proposed as a way to extend SRAM-based FPGA performance to overcome reliability degradation due to Negative Bias Temperature Instability (NBTI) effects [4]. NBTI stress can cause changes in threshold voltage that result in propagation delays similar to those mentioned above. The wear leveling technique described for SRAM-based FPGAs is to periodically change the resource allocation so that effects of degradation are mitigated. This same general principal can be applied to flash-based FPGAs as well. Wear leveling is a well-established standard requirement for large density flash-based memory devices. There are three main wear leveling strategies [4]:

1. Alternative Logic Mapping—Periodically inverting the sense of nets in a design so that a single level does not dominate the signals.
2. Spare Resources—Timing critical functions are moved around an asset of surplus resources. When unused these resources are configured to a “low-degradation” state.
3. Alternate Placements—Configurations for a given design are assigned different placements so that each resource is stressed under two or more different functions.

Wear leveling is usually implemented as part of the overall design strategy from the beginning. This means there is no need for the FPGA to perform self-test or some other computation in the field of operation to determine when degradation may be beginning to negatively affect a circuit element or node. An example of the improvement implementing wear leveling can provide is shown in Figure 3.2-1. Here a
spare resource approach was used on the input A of an XOR function under a variety of stress conditions. It can be seen that for both the failing edge and rising edge timing conditions the improvement can be approximately 20% to 25% for the worst-case scenarios. The entire set of experiments described by [4] showed that in an accelerated life test condition that emphasizes NBTI degradation an improvement of 1.9X in lifetime could be obtained. This improvement in lifetime is defined as the time it takes for a given propagation degradation to occur. The benefits of adopting this type of approach to enhancing long-term stability of FPGAs for space applications are clear.

![Impact of spare-resource wear leveling on input of XOR function](image)

**Figure 3.2-1.** Impact of spare-resource wear leveling on input of XOR function. [4]

### 3.3 Radiation Related Effects

The radiation performance (and limitations) of the ProASIC line of flash FPGAs has been well documented [5]. Discussion of these effects is provided in this report as part of the overall failure mechanism description and subsequent mitigation strategy. The idiosyncrasies of the flash cell remain the main technology limiter to implementing this technology, so the more general and encompassing the mitigation efforts the larger the impact on final device implementation and operation. Radiation testing gives particular precise insight into failures mechanisms of the flash cell.

High dose rate TID testing showed that the ProASIC core could with stand levels to about 20krad before failure occurred. Radiation-induced mechanisms can be divided into holes injected into the floating gate, holes trapped in the oxides, and electrons that become trapped at the polysilicon/oxide interface.

As shown in Figure 3.3-1 the effect of TID radiation on the flash cells is nonuniform. Programmed cells have a much smaller amount of threshold voltage degradation for the same amount of TID exposure than do the erased cells. The erased cells drift almost 2V while the programmed cells only drift about 50mV. Charge loss on the programmed cell with the floating gate containing a large number of electrons is reduced when compared to the erased cell that has had most of the free electrons driven off the floating gate.
Radiation combined with excessive program and erase cycling would cause the erase state-related failures to occur first in this example. This needs to be translated into a specific type of rising edge or failing edge type of failure that can be monitored on sensitive test structures. This type of characteristic signature would be part an overall qualification plan to provide designers with system-level insight into particular failure cases.

The tunnel oxide is degraded every time electrons tunnel through it. A relationship between the amount of charge needed to break down a tunneling oxide and the number of program/erase cycles is described in equation 2. The amount of charge needed to break down the oxide for a given number of program/erase cycles can be scaled from the amount of charge injected as a result a radiation strike.

\[
N_{P/E} = \frac{Q_{bd} A_{inj}}{V_{gg} C_{tun}} \quad (2)
\]

Equation 2 shows that 10,000 program/erase cycles may be equivalent to 20krad TID dose, for example. Such a condition would obviously limit the TID performance of the device. TID effects in flash FPGAs have been shown to have a strong annealing effect as well. This effect can be used to obtain some limited (10%–20%) increase in overall resistance to TID dose. But the program/erase history of the device needs to be considered to ensure that the expected performance can, in fact, be obtained.
4.0 SUMMARY

Flash-based FPGAs from Microsemi offer a unique combination of reprogrammability and nonvolatile performance. The FPGA fabric lends itself to a low power capability that is highly valued by both commercial as well as space applications. However, the use of a flash cell has inherent limitations. These limitations center on the variability that is a natural result of the program/erase process, along with a well-documented, relatively soft TID dose response. Because the flash cells are, by definition, the basis for the entire FPGA interconnect scheme and logic resource, the lack of an redundant or error-correcting scheme similar to a conventional flash memory make the part susceptible to timing-sensitive degradation.

The qualification work done to date matches historical enhanced MIL-STD class B flows with burn in and life test. As documented in the FY11 report, the flash ProASIC FPGAs have passed this type of qualification. The concerns highlighted in this report reflect the lack of standardized testing and screening that takes flash-cell-unique failure modes into account. Currently, as defined by Microsemi, only one pattern is programmed and erased into the device prior to extended life test and burn in. This type of test would only re-enforce preferred states of flash cells and does not cover possible disturbance-related upsets that might occur when updates to designs are being loaded and evaluated.

This regime needs to be extended to more closely look at a variety of different FPGA designs and resource utilizations as a function of radiation and program and erase conditions. Particular sensitivity to possible endurance and disturbance-related failures in even a low to moderate radiation environment have not been evaluated. This type of typical mission scenario represents a complex series of possible interactions and degradations that are, together, truly convoluted.

A variety of wear leveling techniques are available to help mitigate and possibly extend high reliability requirement mission life. This flash FPGA technology needs to be further examined to resolve this detail issues and provide detailed mission guidance.
5.0 REFERENCES


