

Flight Control Fault Models Based on SEU Emulation

Christelle Hobeika, Simon Pichette, Azeddine Ghodbane, Claude Thibeault, Yves Audet,
Jean-François Bolland, and Maarouf Saad
Ecole de Technologie Supérieure

ABSTRACT

The increased use of FPGAs over the past decade has induced an increased concern about radiation effects, in particular the effects of single event upsets (SEU) in SRAM-based FPGAs. Technology scaling and density increase have caused FPGAs to be more vulnerable to SEU. Therefore, external radiations present an issue not only for space based systems; but also for critical terrestrial applications operating in harsh environment, such as commercial avionics.

In order to build robust fault tolerant systems, SEU effects have to be analyzed and modeled so that the designer understands and considers the system's possible faulty behaviors. In this paper, we present a complete automated methodology, based on the use of SEU controller provided by Xilinx, to efficiently emulate SEUs on an FPGA design and extract possible fault models based on radiation effects. The proposed method is applied on a reconfigurable flight control system based on a reference adaptive control model. With the automatic aspect of the proposed methodology, it was possible to emulate a large number of SEUs with reduced time and effort. Experimental results present the design sensitivity, its failure rate as well as its faulty output behavior. Moreover, results analysis disclose the existence of new actuator control fault models that are not considered in fault tolerant systems, these systems being mainly based on the existing well-known models in the literature. The new fault models can be used by the designer at an earlier stage in the design to build robust flight control systems.

CITATION: Hobeika, C., Pichette, S., Ghodbane, A., Thibeault, C. et al., "Flight Control Fault Models Based on SEU Emulation," *SAE Int. J. Aerosp.* 6(2):2013, doi:10.4271/2013-01-2246.

INTRODUCTION

The Field Programmable Gate Array (FPGA) has found its way into avionics and aerospace industries, due to its reconfigurable aspect, low non-recurrent engineering costs and high performance. Besides, system level functional changes can be implemented, reprogrammed and transmitted remotely, which makes it uniquely suited for remote missions.

However, radiation effects on semiconductor are a major concern in avionics and space applications. In fact, highly energetic primary solar and galactic cosmic rays generate secondary particles when interacting with atoms in the earth's atmosphere. Each particle can interact with Si within microelectronic devices resulting in an impulse charge that may initially affect the local circuit and then propagate throughout the system under some conditions causing serious consequences such as loss of information, functional failure or loss of control. These effects on semiconductor devices, called single event effect (SEE), have been an area of interest

since upsets were observed in space and avionic applications several years ago. SEE causing electronics within aircraft systems to malfunction during flight have been observed and documented in [7].

FPGA vendors have provided radiation hardened FPGAs for use in space. These devices are much more expensive than contemporary commercial products and are few technology nodes behind. Therefore, the use of commercial SRAM-based FPGA has increased and heightened concerns about radiation effects, in particular the effects of single event upsets in SRAM-based FPGAs [1]. These FPGAs are based on static memory (SRAM), where the whole configuration defining the design functionality and routing is stored. The energy deposited by the radiation particles interacting within the device can result in a variety of different effects. The most probable effect when considering SRAM-based FPGA is single event upsets (SEU). An SEU is a change of state in a memory that causes a bit flip in the memory cell. Therefore, SEUs in an SRAM-based FPGA can alter the functionality or

the routing of the implemented design and result in a faulty behavior that remains until a new device reconfiguration.

Many techniques have been developed to ensure a safe and correct behavior despite the SEU in SRAM-based FPGA. Some of these techniques, such as triple modular redundancy (TMR) with voters [8] ensure immunity to SEU and must be modeled and implemented in the design itself. Other techniques, like error detection and correction (EDAC) [9], are used to protect data stored in memory cell, the basis is a decoding/encoding algorithms that can be done in the hardware or software to restore the correct values. In order to build a robust radiation tolerant system and implement any of these techniques, we first need to determine, for each sub module to be implemented in an FPGA, its sensitivity to SEUs, its operation failure rate and most importantly its faulty behavior in presence of SEUs.

Aircraft fault tolerant control systems can detect and compensate for many types of faults, such as actuator faults for example, to avoid system failures; the detection and diagnosis of faults are based on a certain set of fault models considered to be the most common. However, these models don't take into account SEU effects, and in order to ensure reliability and robustness of the system we need to analyze the system faulty behavior in the presence of SEUs.

The most accurate way to do this is to perform hardware testing [10]. In this technique, the device implementing the design is exposed to particle bombardment where a particle accelerator is used for characterizing heavy ion and proton inducing single event effects on the device. This technique is very expensive and doesn't present any experiment controllability to where and when the SEU is induced or repeatability. Several works have been developed for sensitivity and soft error estimation, as analytical techniques [4, 5, 6] or fault injection techniques that can be applied at early stages of the design at minimal cost. Moreover, fault emulation can be used to inject errors in a controlled and predictable way on FPGA configuration memory, instead of performing particle bombardment. This can be done using the SEU controller IP core provided by Xilinx [5]. The basic concept is to emulate SEUs as if the device is exposed to radiation. Emulation is performed on a design at a low level; it insures accuracy, controllability and repeatability. However, a large number of experiments have to be done efficiently in order to get accurate results; therefore emulation and results acquisition and analysis can be time consuming and very complex.

In this paper we present a detailed description of a complete automated methodology to extract a design fault models based on SEU effects, using the SEU controller emulation. The proposed method is applied on a reconfigurable flight control system in order to model its faulty behavior in the presence of SEUs.

The paper is organized as follows. First, we describe fault tolerant control systems and the existing control fault models. The next section outlines SEU effects on SRAM-based FPGA, and then we present the complete automated

methodology to efficiently emulate SEUs with reduced time and effort. Experimental results describe the design sensitivity to SEUs and present extracted fault models. Finally, several remarks conclude the paper.

FAULT TOLERANT CONTROL SYSTEMS

Flight controls have advanced considerably through the years, resulting in more complex and sophisticated systems and higher aircraft performances. Electronics was first introduced into flight control to overcome stability issues like the Dutch roll, then the technology advanced and rapid development of digital electronics led to digital "fly-by-wire" systems in which mechanical linkage between the pilot and flight control actuators is removed and the aircraft control relies totally on electronic means to control the aircraft, as described in Fig. 1. Due to the weight saving and considerable improvements most modern fighter aircraft possess a fly-by-wire system; it has been applied to civil transports as well [13].

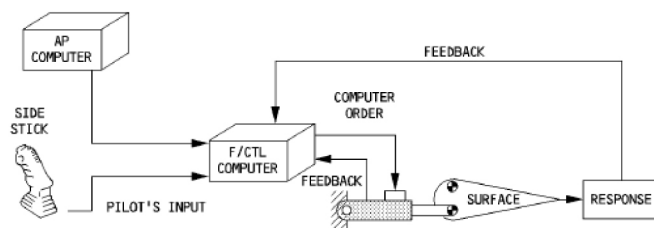


Figure 1. Fly By Wire, Source: <http://www.ahrtp.com>

Cosmic rays induced faults can affect aircraft stability and degrades performances by affecting control signals. The aforementioned faults due to SEUs have to be taken in consideration when system failure is considered, in addition to all other faults. In 2008, an Australian aircraft was cruising at 37,000 ft, when it abruptly pitched nose down twice, due to a sudden change in the position of the aircraft's elevators. Each time, after about 2 seconds, the aircraft responded normally to the pilot's sidestick input. 110 of the 303 passengers were injured. According to the report of the ASIR (Aviation Safety Investigation and Reports), it was concluded that the failure was related to cosmic rays [15]. Radiation effects can be a main concern of the fly-by-wire systems where actuator control relies on digital electronics. An actuator failure can cause damages at several levels since it is used to deflect control surfaces such as ailerons, elevators and rudders and to actuate the engine throttle or the landing-gear mechanism.

In order to ensure the system's reliability and safety, fault tolerant control systems are developed to undertake high integrity control tasks and guard against equipment failure. These systems can detect and compensate for many types of faults, including actuator faults control. There are the passive fault tolerant controllers that compensate for the actuator deviations from their expected position by a fixed robust

feedback controller [11], and the active fault tolerant controllers that have a separate module, the Fault Detection and Diagnosis FDD process that monitors the health of the aircraft and may decide to reconfigure the flight controllers, the guidance system and the navigation system [12]. FDD processes are based on the most common actuator control fault models seen in the literature which include: floating around trim, locked-in-place, hard-over and loss of effectiveness, as described in Fig. 2[4].

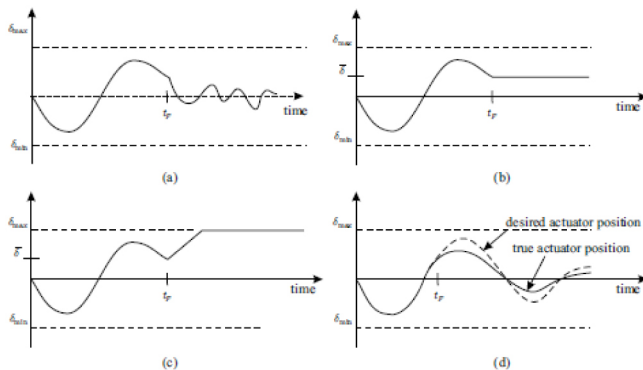


Figure 2. Actuator fault models [4]. a) Floating around trim, (b) Locked-in-place, (c) Hard-over, (d) Loss of effectiveness.

Here, we try to check if these fault models take into account actuator control fault behavior in the presence of SEUs. The system used for this purpose is described in Fig. 3. It is a flight control system based on a reference adaptive control model approach (MRAC) [4]. This technique is often used as a final stage of a complex control system combining several algorithms. The goal is to have the output of the plant under consideration follow the output of a reference model by adapting the controller parameters. The controller and the adjustment mechanism represent the reconfigurable flight control (RFC) system, that produces the control signal $u(t)$. The RFC will be bombarded by SEUs and its output $u(t)$ will be monitored to detect any disturbance and faulty behavior. Therefore, we present a complete automated methodology to efficiently perform a large number of experiments with reduced time and effort in order to extract fault models based on SEU. This methodology will be applied on the control system in Fig. 3.

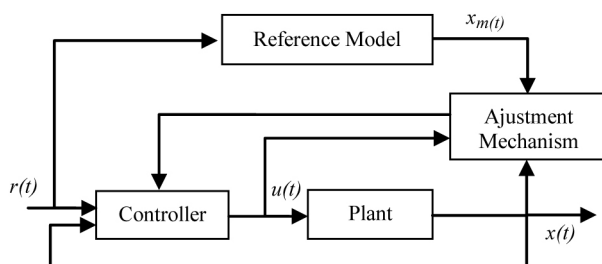


Figure 3. Model Reference Adaptive Control Scheme [14].

SEU EFFECTS ON SRAM-BASED FPGAS

An SRAM based FPGAs is a re-programmable logic device, that contains programmable I/O blocks, memory modules, as well as flexible logic and routing resources; where the logic configuration is stored in SRAM cells called configuration bits. These values are loaded in the memory on power-up and can be reloaded to change the device functionality. The abundance of routing resources permits the accommodation of large and complex designs. As mentioned before, this type of device is very sensitive to radiations: if a particle penetrates the susceptible portion of a configuration SRAM cell it alters its state resulting in an SEU and thus a configuration bit flip. Therefore, an upset in the SRAM has the potential to change the design definition (routing, functional logic etc.) which may lead to an operational failure.

A typical FPGA design only involves a fraction of the total number of configuration memory cells. These bits are associated with the design circuitry and are called essential bits. The memory bits associated with the unused portion of the device are not essential. If an essential bit is upset it will change the design circuitry. However the upset may or may not affect the function of the design. If it does cause a failure and affect the operation, it is considered a critical bit. Therefore, only a subset of the essential bits that define the design can cause a failure and the rate at which the device can experience SEU is not the same as the rate at which the operation will be affected. It takes from 10 to 100 upsets to actually cause a functional failure [2].

Emulating an SEU in an SRAM-based FPGA is done by flipping a bit in its configuration memory. The emulation can be done at a specific well known location in the configuration memory with the use of a logicCore IP provided by Xilinx, the SEU controller [6]. Design behavior should be monitored to see when and how its operation is disturbed and deduce design sensitivity.

The SEU controller macro provides a convenient way to emulate SEUs within the device by injecting errors in a controlled and predictable way into the configuration memory. It should be implemented within the device and can be controlled via a UART interface. However, the use of SEU controller requires much time and effort, especially if a large number of samples is needed to evaluate upset effects. An efficient methodology is needed to build a statistical profile of how susceptible a design is to an SEU within a given device and how the device behaves when it is affected.

METHODOLOGY

Our goal is to evaluate and quantify SEU effects by performing SEU emulation on an SRAM-based FPGA implementing a design, observe its faulty behavior in the presence of SEU and extract the corresponding fault models. Here we present an automated methodology to efficiently use