Analysis of System-Failure Rate Caused by Soft-Errors using a UML-Based Systematic Methodology in an SoC

Analysis of System-Failure Rate Caused by Soft-Errors using a UML-Based Systematic Methodology in an SoC

Mohammad Hosseinabady,
M. H. Neishaburi, Zainalabedin Navabi
University of Tehran, Iran
{mohammad, mhnisha}@cad.ece.ut.ac.ir,
navabi@ece.neu.edu

Alfredo Benso,Stefano Di Carlo,
Paolo Prinetto
Politecnico di Torino, Italy
{ alfredo.benso, stefano.dicarlo,
paolo.prinetto}@polito.it

Giorgio Di Natale
LIRMM,
Université Montpellier II /
CNRS UMR 5506, France
giorgio.dinatale@lirmm.fr

Abstract: This paper proposes an analytical method to assess the soft-error rate (SER) in the early stages of a System-on-Chip (SoC) platform-based design methodology. The proposed method gets an executable UML (Unified Modeling Language) model of the SoC and the raw soft-error rate of different parts of the platform as its inputs. Soft-errors on the design are modeled by disturbances on the value of attributes in the classes of the UML model and disturbances on opcodes of software cores. The Dynamic behavior of each core is used to determine the propagation probability of each variable disturbance to the core outputs. Furthermore, the SER and the execution time of each core in the SoC and a Failure Modes and Effects Analysis (FMEA) that determines the severity of each failure mode in the SoC are used to compute the System-Failure Rate (SFR) of the SoC.

1 Introduction

Low cost and high-performance System-on-Chips (SoCs) are easily manufactured but often without satisfying the requirements of dependable computing. In fact, the more the process technology scales and the feature sizes shrink, the more the circuits become susceptible to transient faults. Transient faults, including those caused by crosstalk, substrate and power supply noise, charge sharing, etc., pose a significant challenge to ensuring signal integrity in deep submicron process technologies. In addition, current studies indicate that circuits will become increasingly sensitive to temporary faults caused by terrestrial cosmic rays and alpha particles, and that this will result in unacceptable soft-error rates (SERs) even in mainstream commercial electronics.

Error protection mechanisms, such as radiation-hardened circuits or architectural redundancy, however, come with significant penalty in performance, power, and area. Consequently, designers must evaluate the system failure rate of a system at early stages of the design process to decide the appropriate amount of protection necessary for the target market.

1.1 Contributions

In this paper we propose a quantitative method to evaluate system failure rate (SFR) at the early stage of an SoC design. The system failure rate caused by a transient noise is computed by using the soft error rate of different cores in the SoC, and the severity of each error in each core on the whole system. The proposed method relies on an executable system level model of the SoC (which is described by UML-RT [1]) to compute the soft-error rate of each module in the system.

Contributions of this paper are briefly listed as follows:

1. Use an executable specification model based on UML-RT (Unified Modeling Language Real Time) as a functional prototype of the SoC to assess soft-error rate;
2. Model a soft-error in the final product by a disturbance in a variable in the class attributes;
3. Propose a probabilistic method to compute the propagation probability of an error on a variable in an algorithm as a measure of soft-error propagation;
4. Consider the hardware core and software core separately during the assessment process;
5. Consider the platform contribution on the dependability of the SoC.

The method proposed in this paper is different from the method that has been proposed in [3]. In this paper, we have considered the architecture vulnerability factor (AVF) of different parts of a system to assess System-Failure Rate (SFR), whereas [3] uses the timing vulnerability factor (TVF) of different parts of a system to assess SFR.

This paper is organized as follows. Our proposed method is explained in Section 2. Section 3 presents the experimental results and Section 4 concludes the paper.

2 Proposed Method

The main objective of this paper is to propose a method to assess the soft-error rate of an SoC. This high level soft-error rate can be used to compare the implementation of the

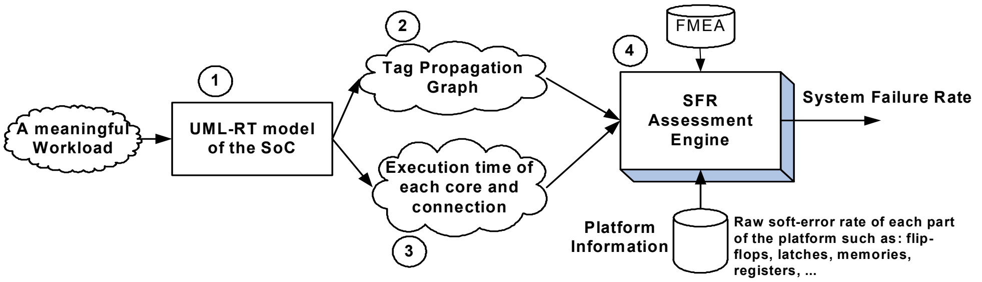

1
2
3
4
A meaningful Workload
UML-RT model of the SoC
Tag Propagation Graph
Execution time of each core and connection
FMEA
SFR Assessment Engine
System Failure Rate
Platform Information
Raw soft-error rate of each part of the platform such as: flip-flops, latches, memories, registers, ...

Figure 1 The proposed methodology

13th IEEE International On-Line Testing Symposium (IOLTS 2007)
0-7695-2918-6/07 $25.00 © 2007 IEEE

IEEE COMPUTER SOCIETY

SoC on different platforms with respect to the obtained dependability, or to analyze the contribution of different parts of the SoC to their impact on the SoC dependability. We use the UML-RT as the specification language to describe the SoC. Furthermore, we assume that the designer has already partitioned the software and hardware parts of the system.

2.1 Proposed fault model

Single Event Upsets (SEUs) occur in storage cells consisting of latches, flip-flops, registers, and memories. Consequently, they cause changes on the data or state of the storage cells.

2.2 Proposed computational method

The proposed method computes the vulnerability of cores in the event of soft-errors in their variables. The dynamic behavior of a core has impact on its vulnerability to soft errors in the core variables. Core vulnerability factor and the propagation probability of variable tags to the core outputs represent this impact. On the other hand, the execution time of a core has impact on the vulnerability of the SoC that is evaluated with the time vulnerability factor.

Figure 1 shows the different steps of our methodology that can be summarized in the following steps:

1. Simulating the UML-RT model of the SoC with a meaningful workload;
2. Constructing a variable dependency graph called **tag propagation graph** for each core during the simulation of the UML-RT model that is used to compute the core vulnerability factor;
3. Monitoring the execution time of each core and connector between cores during the simulation that is used to compute the time vulnerability factor of each core and connector of the SoC;
4. By using the raw soft-error rate (soft error rate in the semiconductor) of storage cells in a platform, the graphs of Step 2, the Failure Mode and the Effect Analysis (FMEA) [2], and the execution times of Step 3, the SFR computation engine first computes the hardware/software core error rate, and then computes the SoC failure rate.

3 Experimental Results

We have applied the proposed algorithm to the JPEG compression system which its structure diagram is shown in Figure 3 [3]. We assign severity indices of 0.25, 0.50, 0.75, and 0.95 to minor, marginal, critical, and catastrophic severity classes, respectively. Figure 2 shows our assumptions for the severity of different parts of the compression system.

Core	Severity
YCBCR	0.50
Blocking	0.50
Downsampler	0.50
FDCT	0.50
Quantizer	0.25
Zigzag	0.50
Huffman	0.75

Connector	Severity
Input$\xrightarrow{\text{to}}$*YCBCR*	0.25
YCBCR→*Blocking*	0.25
Blocking→*Downsampler*	0.25
Downsampler→*FDCT*	0.25
FDCT→ *Quantizer*	0.25
Quantizer →*Zigzag*	0.25
Zigzag→*Huffman*	0.25
Huffman→ *output*	0.95

Figure 2 Severity of cores and connectors

We also assume that all cores in the JPEG system except the DCT core are software cores. To evaluate our method, we have injected SEU transient faults in different cores of our UML-RT model. We have also observed the output of the system in the presence of each fault to assess the erroneous effect of that fault.

Figure 4-*a* shows the average percentage of erroneous effect on the fault images caused by injected faults in different parts of the system with respect to the original images. To compute these percentages, we have evaluated the percentage of the corruption in the resulted picture during the fault injection process in the presence of SEUs in different parts. Using the proposed method, Figure 4-*b* shows the contribution of the core errors to the whole system error rate. Based on this diagram, the Huffman-coding core has the most contribution in the reliability of the JPEG system. Comparing Figure 4-*a* and Figure 4-*b* confirms the validity of the proposed method.

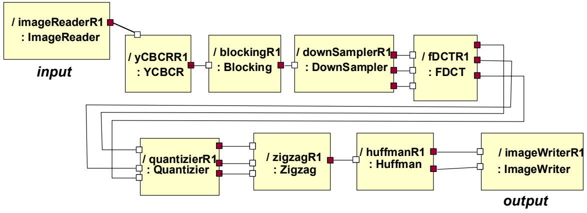
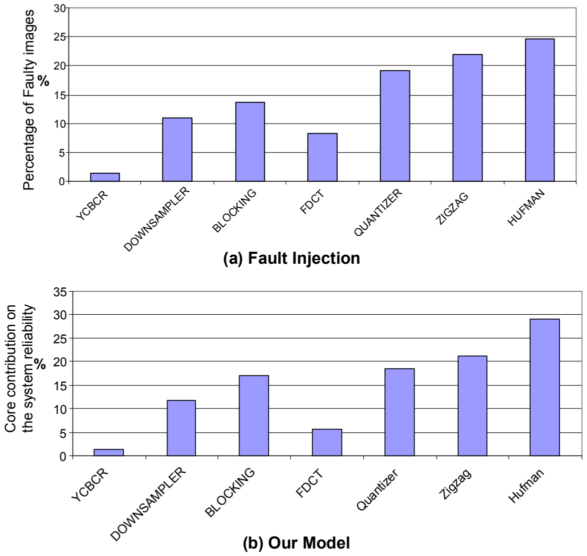


Figure 3 Structure diagram of JPEG system



(a) Fault Injection

(b) Our Model

Figure 4 Experimental results

4 Conclusions

This paper proposes an automatic analytical soft-error rate assessment for a System-on-Chip (SoC) designed. The proposed method is based on an executable UML-RT model of the SoC. The method processes simultaneously the algorithm specification, the characteristics of the chosen platform, and the corresponding fault information.

References:

[1] Rational Rose® RealTime, "Modeling Language Guide," Version 2003.06.00, *http://www.rational.com*.

[2] G. S. Wasserman, "*Reliability Verification, Testing, and Analysis in Engineering Design*," New York, NY, USA: Marcel Dekker Incorporated, 2002.

[3] M. Hosseinabady, M. H., Neishaburi, P. Lotfi-Kamran, and Z. Navabi, "A UML Based System Level Failure Rate Assessment Technique for SoC Designs," *to be appear in VTS07.*