



FP7-ICT / FET-OPEN – 309129 / i-RISC

D2.3

Energy models of sub-powered CMOS circuits

Editor:	Alexandru Amaricai (UPT)
Deliverable nature:	Public
Due date:	July 31, 2015
Delivery date:	August 31, 2015
Version	2.0
Date of current version:	April 15, 2016
Total number of pages:	37
Reviewed by:	i-RISC partners
Keywords:	Energy modeling of faulty CMOS circuits, Energy modeling of interconnects, Simulated fault injection

Abstract

This report presents an overview of the activities carried out in the Work Package 2 (WP2) framework the third i-RISC reporting period. The main contributions included in this deliverable are: a hierarchical methodology for energy modeling of faulty CMOS circuits (Tasks 2.1, 2.5) and a simulation based energy evaluation framework for on-chip interconnects (Task 2.1, 2.5). Furthermore, developments and enhancements of the multi-level simulated fault injection for probabilistic timing errors, and probabilistic FPGA fault emulation (Task 2.4) are included.

List of Authors

Participant	Author
TU-Delft	Nicoleta Cucu-Laurenciu
	Sorin Cotofana
	Joyan Chen
UPT	Alexandru Amaricai
	Sergiu Nimara
	Flavius Pater
	Ioana Mot
	Dan Dutescu
	Oana Boncalo
CEA	Valentin Savin

Table of Contents

List of Authors	2
Table of Contents	3
List of Figures	4
List of Tables	5
Abbreviations	6
1. Executive Summary	8
2. Energy Modeling of Faulty Logic Circuits	. 10
2.1. Energy model for faulty circuits	. 10
2.2. Single transition-based energy model of faulty circuits	. 10
2.3. Multiple transition-based energy model of fault free circuits	. 17
2.4. Conclusion	. 22
3. Interconnects Energy Measurement	. 23
3.1. Simulation Framework and Methodology	. 23
3.2. Simulation Results	. 24
3.3. Conclusion	. 26
4. Performance Enhancement of Multi-Level Fault Injection and Emulation	. 27
 4.1. Multi-Level Probabilistic Timing Error Reliability Analysis 4.1.1. Related Work 4.1.2. Proposed Methodology 4.1.3. Case Study 	. 27 28 29 30
 4.2. Performance enhancement of serial based FPGA probabilistic fault emulation techniques 4.2.1. Serial FPGA emulation scheme	. 31 31 31 32
4.3. Conclusion	. 34
5. General Work Package 2 Conclusions	. 35
References	. 37

List of Figures

Figure 1-1 - WP2 Gantt Diagram	8
Figure 2-1 Adjuvant Energy Measurement Explanation for a Gate	11
Figure 2-2 3-Input Majority Circuit	14
Figure 2-3 Majority Circuit <i>EF</i> Variation with the Primary Inputs Transition	16
Figure 2-4 Majority Circuit E Variation with the Primary Inputs Transition.	16
Figure 2-5 PCA Components Loadings	18
Figure 2-6 PCA Components Dataset Variance	18
Figure 2-7 Energy Approximation for $(M, N) = (100, 20)$	21
Figure 2-8 Energy Approximation for $(M, N) = (1000, 20)$.	21
Figure 2-9 Energy and Output Statistics Approximation for $(M, N) = (100, 20)$	22
Figure 2-10 Energy and Output Statistics Approximation for $(M, N) = (1000, 20)$	22
Figure 3-1 Microstrip Sectional View.	23
Figure 3-2 Stripline Sectional View	23
Figure 3-3 Interconnect Simulation Flow.	24
Figure 3-4 n Conductor Line Model [Synopsys]	24
Figure 3-5 Simulation Setup	24
Figure 3-6 Wire Length vs. Energy Gain for the Coded 9-Wire Bus	25
Figure 3-7 Wire Length vs. Energy Consumed	25
Figure 3-8 Box-and Whisker Plot Convention	26
Figure 3-9 Data Arrival Profile for 8-Wire Bus	26
Figure 3-10 Data Arrival Profile for 9-Wire Bus	26
Figure 4-1 – Three level reliability analysis methodology	29
Figure 4-2 – Performance improved serial FPGA fault emulation scheme	32
Figure 4-3 – Insertion and emulation phase for the serial FPGA fault emulation scheme (a) and	
modified serial FPGA fault emulation scheme (b)	33

List of Tables

Table 2-1 NAND2 - Fault-Free Gate Energy Consumption	. 12
Table 2-2 NAND2 - Faulty Gate Energy Consumption	. 12
Table 2-3 XOR2 Fault-Free Gate Energy Consumption	. 13
Table 2-4 XOR2 Faulty Gate Energy Consumption	. 13
Table 2-5 Faulty Majority Circuit	. 15
Table 2-6 Gate Input and Output Statistics	. 19
Table 2-7 Training and Testing Energy Approximation Results	. 20
Table 2-8 Training and Testing Energy and Output Statistics Approximation Results	. 21
Table 2-9 Training and Testing Energy and Output Statistics Approximation Error	. 21
Table 4-1: Implementation Results for Fault Injected Barrel Shifter	. 33

Abbreviations

BRAM	Block Random Access Memory
CMOS	Complementary Metal Oxide Semiconductor
CUT	Circuit Under Test
DFF	D Flip-Flop
DEMUX	De-multiplexer
DUT	Design Under Test
ECC	Error Correcting Code
EFI	Emulated Fault Injection
FER	Frame Error Rate
FO	Fan-Out
FPGA	Field Programmable Gate Array
GD	Gaussian Distribution
GF	Galois Field
GL	Gate Level
HDL	Hardware Description Language
IG	Inverse Gaussian distribution
ICON	Integrated Controller
ILA	Integrated Logic Analyzer
LDPC	Low Density Parity Code
LFSR	Left Feedback Shift Register
LUT	Look-Up Table
MAE	Mean Absolute Error
MAJ	Majority Voter
MCS	Monte Carlo Simulation
MSE	Mean Squared Error
MUX	Multiplexer
NN	Neural Network
OCV	On-Chip Variability
PCA	Principal Component Analysis
PDF	Probability Density Function
PRNG	Pseudo Random Number Generator
PVT	Process Voltage Temperature
RLC	Resistance, Inductance and Capacitance
RNG	Random Number Generator

RTLRegister Transfer LevelSEUSingle Event UpsetSFISimulated Fault InjectionSNRSignal to Noise RatioTRNGTrue Random Number GeneratorVIOVirtual Input / OutputWPWork Package

1. Executive Summary

This deliverable reports the developments associated with Work Package 2 (WP2) during the third reporting period (M22-M30), which cover the following main research avenues: (i) the hierarchical energy modeling of faulty CMOS circuits and (ii) the energy evaluation of in-chip interconnects. Furthermore, it includes the latest achievements related to probabilistic fault injection and FPGA fault emulation for sub-powered CMOS circuits.

WP2: FALLET MODELS / ENERGY MEASURES				YEAR 1				YEAR 2			YEAR 3		
•••	2.1 AOE1 WODELS / ENERGI WEAGORED												
	Deliverables				2.1			2.2			2.3		
	T2.1: SPICE analysis for sub-powered circuits		1	1									
s	T2.2: Fault models for uncorrelated errors		1	1									
ask	T2.3: Fault models for correlated errors												
L	T2.4: Higher abstraction levels & fault inject.												
	T2.5: Energy models for sub-powered circuits												

Figure 1-1 - WP2 Gantt Diagram

The main contributions associated to this deliverables are related to Task 2.1, Task 2.5, and Task 2.4, as follows:

- Hierarchical energy modeling of faulty CMOS digital circuits (Task 2.1, Task 2.5) We propose a bottom-up, composite, energy assessment framework, which takes into account the failure probability of individual logic gates due to various fault-inducing factors (e.g., technology variability, environmental aggression). We developed accurate and effective, in terms of computational resources, energy evaluation methods for the following cases: (i) single transition based energy modeling and (ii) multiple-transitions based energy consumption evaluation. Our approach provides the necessary means for accurate energy evaluation, required in the WP3 and WP4 framework for the assessment of energy-error correction capability-throughput tradeoffs provided by different LDPC decoding algorithms.
- Energy evaluation methodology for on-chip interconnects (Task 2.1, Task 2.5) We present
 a simulation methodology able to evaluate interconnects performance in terms of energy
 consumption and propagation delay. The aim inhere is to provide the means for the
 evaluation and comparison of different interconnect coding techniques introduced in WP4.
 The proposed methodology relies on an electro-magnetic solver and on SPICE based
 simulations, and derives both energy and propagation delay measurements for different
 interconnect configurations.
- Developments in probabilistic simulated fault injection and FPGA fault emulation (Task 2.4)

 We present a hierarchical, three layers, fault injection technique, targeting the reliability assessment of probabilistic timing errors. The proposed methodology relies on statistical static timing analysis performed for standard cell components SPICE models, probability density function propagation at gate level, and simulated fault injection performed at Register Transfer Level (RTL). We further propose an FPGA fault emulation performance

enhancement of the serial based FPGA probabilistic fault emulation scheme we previously proposed in Deliverable D2.2.

The deliverable is organized as follows: Chapter 2 presents the faulty logic CMOS circuits hierarchical energy modeling approach and framework. Chapter 3 is dedicated to the on-chip interconnect simulation based evaluation methodology while our recent results related to the probabilistic simulated fault injection and fault emulation are included in Chapter 4. The general WP 2 conclusions are presented in Chapter 5.

2. Energy Modeling of Faulty Logic Circuits

Abstract: In the reign of CMOS deep-submicron devices variability, the behavioral predictability decreases with each technology node. Due to unpredictable devices a nonzero probability exists (error rate) that expected results are not delivered by certain logic gates within a computation unit implementation. Depending on the circuit topology and evaluated input data the fact that a certain gate produces a wrong output may or may not affect the circuit primary outputs values but certainly has a direct impact on the circuit energy consumption as it results in a different switching activity within the circuit. State of the art energy consumption estimation methods and tools do not consider such phenomena, which makes them less accurate for circuits built out of unpredictable devices, which is the case within the i-RISC project framework. In view of this we introduce in this section a novel, composite, bottom-up, circuit-level energy assessment framework able to take into account the failure probability of individual comprising gates due to various fault-inducing factors (e.g., technology variability, environmental aggression). We propose accurate and computational effective approaches for the following two cases: single transition based energy modeling and multiple-transitions based energy evaluation. Given that this framework provides the means for accurate energy evaluation of circuits operating in faulty conditions it can be utilized in the assessment of energy-performance tradeoffs provided by different LDPC decoder algorithms and architectures.

Publications: Unpublished work (to be submitted to Microelectronic Reliability).

2.1. Energy model for faulty circuits

We first introduce a general model to assess the energy consumed by a component (e.g., gate, circuit) for a specific input transition/workload and probability that it produces a faulty output. More precisely, we evaluate the component energy consumption as a function of (i) the energy consumed by the fault free component E_{FF} , (ii) the energy consumed by the faulty component E_F , (iii) the probability that the component is producing a faulty output (i.e., the component output logic level is flipped with regard to its correct, error free counterpart, under the same input and environmental stimuli) P_F as follows:

$$E = (1 - P_F) \cdot E_{FF} + P_F \cdot E_F \tag{2-1}$$

We note that P_F value is determined by fabrication technology, aging, and environmental conditions; while E_{FF} and E_F depend of component function and input vector transition.

Given the model in Equation (2-1) we first detail the energy evaluation methodology for a single gate, for a specific transition. Subsequently, we extend the approach from the gate level to the circuit level, and exemplifying it on a 3-input majority circuit (a typical LDPC component).

2.2. Single transition-based energy model of faulty circuits

Let us assume that we want to evaluate the energy of a circuit under a specific workload and environmental conditions, when the circuit is affected by a fault (its primary correct output is flipped as a result of a single faulty gate).

The flow to evaluate the energy of a faulty circuit for a given workload, based on the energy of comprising gates consists of two steps: (i) an a-priori characterization of each gate type form the energy point of view (both E_{FF} and E_F) for all possible input transitions, and (ii) based on the circuit topology and the tabulated energy values for each gate, determine circuit E_{FF} as a sum of comprising gates E_{FF} , E_F by taking into account all possible single faulty gates scenarios), and finally the composite circuit energy E according to Equation (2-1).

Equation (2-2) allows for the evaluation of a gate energy E for a given input transition and probability of faulty output P_F . The energy figures for any basic Boolean gate, i.e., E_{FF} and E_F , can obtained by means of SPICE simulation; in the sequel we made use of a commercial 45nm CMOS technology, for normal operating conditions (supply voltage = 1 V, temperature = 25 °C), and FO4 loading. The energy, for both fault-free and faulty component cases, is measured in SPICE using the supply current integrated over one clock cycle, that is $T_{clk} = T_3 - T_2$ in Figure 2-1:

$$E = V_{VDD} \cdot \int_{0}^{T_{clk}} I_{VDD}(t) dt = V_{VDD} \cdot \int_{T_2}^{T_3} I_{VDD}(t) dt$$
(2-2)

Since we measure the power supply current, the measured energy values reflect both the static (leakage) and the dynamic components of the energy.



Figure 2-1 Adjuvant Energy Measurement Explanation for a Gate

For illustration purpose and given that they are two ubiquitous gates in LDPC decoders we summarize the energy values E_{FF} , E_F afferent to each possible transition for a 2-input NAND (NAND2) gate and a 2-input XOR gate in Table 2-1 and Table 2-2, and in Table 2-3 and Table 2-4, respectively.

We note that we treated fault free and faulty gates in different ways as follows: (i) for the fault free gates we simulated all possible input transitions, i.e., 16 combinations for 2-input gates, (ii) for the faulty gates we collapsed the analysis to the 4 faulty output cases by concentrating on what the input situation is at T_2 regardless of the T_1 status. The decision to carry on a less accurate energy evaluation for faulty gates relates to the fact that E_F impacts to a lesser extent the total energy values (its contribution is modulated by P_F which is rather small to current technology node) thus the extra accuracy will not be visible in the final energy values. We note that the E_F computation accuracy

does not affect the generality of the model in Equation (2-3) and that if needed, i.e., if P_F increases, the faulty gates can be also treated in the same way the fault free gates are.

NAND2 - Fault Free												
State	@ T ₁	State	e @ T₂	F [I]	Output [V]							
in_1	in_2	in_1	in_2	<i>L_{FF}</i> [J]	Output [v]							
		0	0	6.40E-15	1							
0	0	0	1	6.51E-15	1							
0	0	1	0	6.00E-15	1							
		1	1	4.94E-16	0							
		0	0	6.96E-15	1							
0	1	0	1	7.31E-15	1							
0	T	1 0		7.80E-15	1							
		1	1	1.44E-15	0							
		0	0	6.81E-15	1							
1	0	0	1	7.93E-15	1							
T	0	1	0	6.58E-15	1							
		1	1	1.33E-15	0							
		0	0	1.58E-14	1							
1	1	0	1	1.59E-14	1							
T	T	1	0	1.59E-14	1							
		1	1	1.01E-15	0							

Table 2-1 NAND2 - Fault-Free Gate Energy Consumption

Table 2-2 NAND2 - Faulty Gate Energy Consumption

NAND2 – Faulty											
State	e @ T₂	E [1]	Output [V]								
in_1	in_2	2 F [9]									
0	0	1.23E-15	0								
0	1	3.41E-16	0								
1	0	8.24E-16	0								
1	1	1.78E-16	1								

XOR2 - Fault Free										
State	@ T ₁	State	@ T₂	F [1]	Output [V]					
in_1	in_2	in_1	in_2	2 FF [J]	output [1]					
		0	0	3.04E-15	0					
0	0	0	1	1.80E-14	1					
U	0	1	0	1.78E-14	1					
		1	1	1.59E-15	0					
		0	0	7.51E-15	0					
0	1	0	1	8.74E-15	1					
0	T	1 0		9.79E-15	1					
		1	1	1.26E-15	0					
		0	0	7.34E-15	0					
1	0	0	1	9.43E-15	1					
T		1	0	8.15E-15	1					
		1	1	9.83E-16	0					
		0	0	9.72E-15	0					
1	1	0 1		1.77E-14	1					
T	T	1	0	1.81E-14	1					
		1	1	1.03E-15	0					

Table 2-3 XOR2 Fault-Free Gate Energy Consumption

Table 2-4 XOR2 Faulty Gate Energy Consumption

XOR2 – faulty										
State	e @ T₂	E - [1]	Output [V]							
in_1	in_2	2 F [9]								
0	0	5.16E-14	1							
0	1	5.29E-13	0							
1	0	1.46E-15	0							
1	1	2.39E-16	1							

In this way we can characterize any 2-input basic gate (e.g., NAND2, XOR2, INV, OR2, NOR2, AND2) for different fabrication technologies and capacitive loading.

While in principle the same method can be applied to basic gates with a larger fain-in the process is getting more complex and time consuming due to the large amount of input combinations that have to be considered into the SPICE simulations. Moreover the faulty gate analysis is also more difficult due to the larger amount of combinations that have to be analyzed as more transistors may induce

the same output error. In view of this and also of the fact that practical implementations are built upon a set of low fain-in basic gates (most of the time not larger then 4 for AND/OR and 2 for XOR) a method is required for the error rate aware energy evaluation extension to the circuit level. Thus, the question we need to address is:

Given a circuit structure built with error rate aware energy characterized basic Boolean gates and a technology, aging, and environmental condition determined fault probability evaluate the energy consumed by the circuit when is inputs are changing from I_1 to I_2 . To evaluate the energy at the circuit level we can apply Equation (2-1) but prior to this we need to evaluate the fault free E_{FF} and the faulty E_F energy consumption for the overall circuit.

When the primary inputs are changing from I_1 to I_2 all bit transitions are propagated by means of logic level simulation through the circuit, such that for each gate transition dependent energy values can be extracted, based on the tabulated, pre-characterized fault-free and faulty gate energy, as follows:

- E_{FF} can be obtained by adding the transition-dependent E_{FF} values of comprising gates according to the simulation log.
- To compute E_F we have to consider the effects of each and every faulty gate in the circuit on the circuit primary outputs iteratively. If a faulty gate results in a faulty circuit primary output, the primary input transition, the faulty gate E_F energy value, and the E_{FF} energy values of the remaining gates for propagated faulty transition are tabulated. The circuit E_F corresponding to each faulty gate is computed by adding the faulty gate E_F and the remaining fault-free gates E_{FF} values. After iterating through all gates, the circuit E_F value for the given circuit primary input transition is derived as the mean of the tabulated circuit E_F energy values (afferent to each faulty gate scenario which results in a faulty circuit output).

We note that we assumed that only one gate can be faulty at a time, which is quite reasonable assumption for the circuits that we are targeting, e.g., 5-input parity, 3-input majority which are typical LDPC decoder parts. The method can be also applied for more than one faulty gate at a time at the expense of a substantial increase of the number of to be evaluated cases but it is out of the scope of the current deliverable.

For the purpose of illustration we employ as discussion vehicle a 3-input majority gate implementation out of 2 AND gates and 2 OR gates, as illustrated in Figure 2-2. We present subsequently the numerical derivation of the E_F and E_{FF} values for the 3-input majority circuit for the case when we would like to evaluate the circuit energy for the primary inputs transition (0 1 1) -> (1 0 1) and a gate error probability (faulty output) $P_F = 10^{-5}$.





Fa	ulty	Gate	e	Fault-Free Gates												
Stat	e	Sta	ate	Sta	ate	Sta	ate	Sta	ate	Sta	ate	Sta	ate	Sta	ate	Faulty circuit
@ T	1	@	T ₂	@	T_1	@	T ₂	@	T ₁	@	T ₂	@	T_1	@	T ₂	output
i_1	i ₂	i_1	i ₂	<i>i</i> ₁	<i>i</i> ₂	<i>i</i> ₁	i ₂	<i>i</i> ₁	<i>i</i> ₂	i_1	<i>i</i> ₂	<i>i</i> ₁	<i>i</i> ₂	<i>i</i> ₁	<i>i</i> ₂	
	G	1			G	2			G	3		G4			F	
0	1	1	0	0	1	1	0	1	1	1	1	0	1	1	1	F
	G	2			G	1			G	3			G	i4		т
0	1	1	0	0	1	1	0	1	1	0	1	0	1	0	0	I
G3			i 3 (G1			G	2			G	i4		Ŧ
1	1	1	1	0	1	1	0	0	1	1	0	0	1	0	0	I
	G	4			G	1			G	2		G3			Ŧ	
0	1	0	1	0	1	1	0	0	1	1	0	1	1	1	1	

Table 2-5 Faulty Majority Circuit

Obviously E_{FF} at the circuit level is the sum of the fault free energies consumed by the 4 gates in the circuit for the induced input transitions as:

$$E_{FF} = E_{FF}^{G1} + E_{FF}^{G2} + E_{FF}^{G3} + E_{FF}^{G4}.$$
(2-4)

To derive E_F a more complex analysis is required. In case the G1 gate is faulty, we do not tabulate any energy value, as it does not result in a faulty 3-input majority circuit output. When G2 is faulty, the majority circuit is also faulty and thus the energy consumed by the circuit in this case (of faulty G2 gate) is given by $E_F^{G2} + E_{FF}^{G1} + E_{FF}^{G3} + E_{FF}^{G4}$ (the sum of E_F of the faulty gate G2 and E_{FF} for the fault-free gates G1, G3, and G4). We note that the energy value for each gate is dependent on its particular input transition. A similar reasoning applies also for the case of faulty G3, in which case the energy is given by $E_F^{G3} + E_{FF}^{G1} + E_{FF}^{G2} + E_{FF}^{G4}$, and for gate G4, with the energy given by $E_F^{G4} + E_{FF}^{G1} + E_{FF}^{G2} + E_{FF}^{G3}$.

The circuit energy when either of the gates is faulty, can then be computed as an average between the circuit energy figures obtained in each of the three faulty gate cases (G2 faulty, G3 faulty, and G4 faulty), as follows:

$$E_F = avg\{(E_F^{G2} + E_{FF}^{G1} + E_{FF}^{G3} + E_{FF}^{G4}), (E_F^{G3} + E_{FF}^{G1} + E_{FF}^{G2} + E_{FF}^{G4}), (E_F^{G4} + E_{FF}^{G1} + E_{FF}^{G2} + E_{FF}^{G3})\}.$$
(2-5)

By substituting Equation (2-4) and Equation (2-5), into Equation (2-1) we can compute the circuit energy *E*, for a faulty output probability of $P_F = 10^{-5}$.



Next State (primary inputs)

Figure 2-3 Majority Circuit E_F Variation with the Primary Inputs Transition.



Next State (primary inputs)

Figure 2-4 Majority Circuit E Variation with the Primary Inputs Transition.

Figure 2-3 depicts the energy consumed by a faulty 3-input majority gate for all possible next logic states of the 3 primary inputs majority circuit, assuming that the primary input state at T₁ is $(i_1, i_2, i_3) = (0,1,1)$. Figure 2-4 graphically illustrates the total energy consumed by the circuit for all possible next logic states of the primary inputs, for a failure probability of $P_F = 10^{-5}$ and the T₁ primary input state being $(i_1, i_2, i_3) = (0,1,1)$.

As the energy consumed by the circuit gates in the faulty cases has relatively the same order of magnitude as the energy consumed by the fault-free gates, the circuit E_F contributes very little to the total E value of the circuit, as it can be also observed in the previous two graphs. However, this

may not be the case for bigger circuits, where the difference between the E_F and E_{FF} values of the subcomponents can be orders of magnitude apart or for technologies and harsh environmental conditions resulting in large gate failure probabilities.

2.3. Multiple transition-based energy model of fault free circuits

If one is interested in evaluating the energy of a faulty circuit under specific workload scenarios one has to employ the previous method in a transition by transition fashion, which may become rather tedious, especially if the circuit under analysis is bigger. Thus, it would be desirable to exploit the workload characteristics and preferably use as little circuit passes as possible, in order to derive the consumed circuit energy. To this end, we introduce a novel fast energy evaluation method, which allows for energy evaluations by propagating input data statistics instead of signal transitions through the circuit. Our approach requires the pre-characterization of basic gates energy and output data statistics as a function of input stimuli statistics, and for that purpose we rely on Neural Networks (NN) based function evaluation. We note that this characterization is only done once per Boolean gate thus it is not inducing any overhead into the energy evaluation process. Then, when a circuit under a specific workload is analyzed, based on the primary inputs workload statistics and each gate pre-characterization (energy and gate output workload as a function of the gate input workload), a single circuit traversal will suffice to determine the energy and output vectors statistics of each gate. The total energy of the circuit is finally computed by summing up the energy values calculated for all circuit gates. Subsequently, for the sake of presentation simplicity we detail the precharacterization flow for a 2-input logic gate.

Let us assume a 2-input NAND gate, with its primary output vector denoted as o_1 and primary input vectors (workload) denoted as i_1 and i_2 , respectively, of N bits each. Since many workload-based statistics can be devised, a Principle Component Analysis (PCA) [Byrne15] was conducted in order to guide the selection of the statistic parameters, which have the greater impact on the gate energy. PCA thus serves as an NN frontend to reduce the dimensionality of the nominal input dataset, and thus to reduce the NN learning time, and avoid over fitting the training data. We examined the following parameters, aiming to capture both spatial and temporal characteristics of the input vectors i_1 and i_2 :

• Total (in both i_1 and i_2) probability of logic '1' occurrence:

$$PH_i = \frac{\#1}{2 \cdot N}$$

where # a denotes the cardinality of a variable a

• Total (in both i_1 and i_2) average switching probability per bit:

$$SW_i = \frac{\#(0 \to 1) + \#(1 \to 0)}{2 \cdot (N-1)}$$

• Cross correlation between the two input vectors

$$CR_i = \frac{2 \cdot \sum (i_1 \wedge i_2)}{N} - 1$$

where $\boldsymbol{\wedge}$ denotes the bit-wise AND operation.

• Convolution of the two input vectors:

$$CV_i = \frac{i_1 \otimes i_2}{2 \cdot N}$$

where \otimes denotes the convolution operation.

• Variable sequence based convolution:

$$CCw_{i1} = \frac{w \otimes i_1}{N/2}; \qquad CCw_{i2} = \frac{w \otimes i_2}{N/2}$$

where w is an L bits sequence (for the current purpose we employed $w = [0 \ 1]$).

• Probability of pairwise concomitant configurations of each kind (00, 01, 10, 11)

$$P00_i = \frac{\#00}{N-1}; \quad P01_i = \frac{\#01}{N-1}; \quad P10_i = \frac{\#10}{N-1}; \quad P11_i = \frac{\#11}{N-1}$$

• Probability of configurations of each kind (00, 01, 10, 11) time-wise

$$Pt00_i = \frac{\#00}{N-1}; Pt01_i = \frac{\#01}{N-1}; Pt10_i = \frac{\#10}{N-1}; Pt11_i = \frac{\#11}{N-1}$$

For the PCA analysis, 1000 samples of 20-bit i_1 and i_2 were generated randomly, based on which the previously 14 enumerated statistics were computed. The PCA dataset consists of a 1000-by-15 matrix, where the first 14 columns correspond to the input workload statistics, while the last column (column 15) corresponds to the consumed energy. The PCA dataset was normalized to the [-1, 1] interval prior to the PCA analysis. Figure 2-6 depicts the percentage of variance of the dataset explained by each principal component, while in Figure 2-5 are illustrated the principal components coefficients (y-axis) obtained from the PCA analysis for each of the principal components (x-axis). The magnitude of the coefficients is an indication of the correlation strength (a value of 0 signifying unexistance of linear correlation).



Figure 2-5 PCA Components Loadings

Figure 2-6 PCA Components Dataset Variance

One may note also that if the dataset analyzed by PCA exhibits outliers, or nonlinear relationships between variables, the PCA analysis may not succeed in exposing the underlying connections. Thus, depending on the variables, the PCA analysis may only roughly guiding the selection of the input statistics variables. Based on the PCA results and convenience from the computational standpoint, the statistics summarized in Table 2-6 were selected for the estimation of the output energy and afferent output statistics.

Statistic	Gate Input Statistics	Gate Output Statistics
Probability of logic '1'	$PH_i = \frac{\#1}{2 \cdot N}$	$PH_o = \frac{\#1}{N}$
Probability of switching	$SW_i = \frac{\#(0 \to 1) + \#(1 \to 0)}{2 \cdot (N-1)}$	$SW_o = \frac{\#(0 \to 1) + \#(1 \to 0)}{N - 1}$
Variable sequence based convolution for i_1	$CCw_{i1} = \frac{w \otimes i_1}{N/2}$	$CCw_o = \frac{w \otimes o_1}{N/2}$
Variable sequence based convolution for i_2	$CCw_{i2} = \frac{w \otimes i_2}{N/2}$	
Probability of '00' time-wise	$Pt00_i = \frac{\#00}{N-1}$	$Pt00_o = \frac{\#00}{N-1}$
Probability of '01' time-wise	$Pt01_i = \frac{\#01}{N-1}$	$Pt01_o = \frac{\#01}{N-1}$
Probability of '10' time-wise	$Pt10_i = \frac{\#10}{N-1}$	$Pt10_o = \frac{\#10}{N-1}$
Probability of '11' time-wise	$Pt11_i = \frac{\#11}{N-1}$	$Pt11_o = \frac{\#11}{N-1}$

Table 2-6 Gate Input and Output Statistics

Having selected the most relevant statistical parameters we now need to create the 2-input NAND gate model able to operate on them rather than on standard Boolean values. As Neural Networks (NN) allow for an automated abstraction of the relationship between outputs and their inducing factors, they exhibit direct beneficial implications over an analytical approach, especially for more complex, non-linear relationships. We employed a 2-layer feed-forward neural network, with 8 inputs (corresponding to the 8 input statistics), 10 neurons in the hidden layer, and 8 neurons in the output layer (corresponding to the consumed energy and 7 output statistics). As concerns the activation functions, we employed a hyperbolic tangent sigmoid function in the hidden layer, and a linear transfer function in the output layer. To derive a dataset for the neural network, *M* samples of *N* random bits per sample for each primary input of the NAND gate, were generated. Based on these samples, the 8 input statistics values that we would like to approximate, the *N*-bit output sequence was first computed. The reference energy values were obtained sample-wise, analyzing the gate input vectors on a transition-by-transition basis, using per se the fault-free gate energy characterization from Section 2. We note that the same neural network framework holds true also for the E_F energy.

As the gate energy and workload statistics values are on different scales, in order to minimize the bias for one input feature over the other within the neural network, as well as speed up the training process, and improve convergence results, prior to applying the dataset to the NN, it was preprocessed as follows: all input and output statistics were normalized to the [-1, 1] interval (min-max normalization), while the energy data was preprocessed by first applying a log 10 operation, followed by min-max normalization to the [-1, 1] interval. The dataset was partitioned into 80% for training, 10% for validation and 10% for testing. For training the NN, the Levenberg-Marquardt algorithm [Seber03] was employed, with a learning rate of 0.4, a momentum constant of 0.8.

We first present the NN approximation results when the NN has a single output (i.e., the energy), and then for the full configuration (i.e., when the NN has 8 to be approximated outputs).

Table 2-7 summarizes the NN results of the training and testing for 2 instances of the number of samples M per primary input (M = 100 samples and M = 1000 samples) of N = 20 bit each. These values of M and N are selected for the purpose of illustration, and for simulation convenience (as they impact directly the training process convergence duration), without restricting their generality (M and N can be chosen with regard to the required workload dimensions). Bigger values for both M and N will affect the pre-characterization time required for training the NN, however they will not impact the actual computation time for evaluating the energy of a circuit.

Parameter	Training		Testing (new samples)	
raidileter	Value	Value	Value	Value
# of samples	100	1000	100	1000
# of bits per sample	20	20	20	20
NN layers sizes	8:10:1	8:10:1	8:10:1	8:10:1
Epochs	58	633		
Slope	0.72	0.63		
Y-Intercept	0.38	0.51		
Correlation	0.8945	0.7931	0.8455	0.7413
MSE	1.8923E-29	3.5024E-15	2.6215E-29	3.8059E-29
MAE	2.9910E-15	4.4997E-15	3.8941E-15	4.7218E-15

Table 2-7 Training and Testing Energy Approximation Results

To quantify the approximation goodness, we employ the following three metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and the correlation between the reference and the approximated energy values. A high correlation is observed in both cases (1 being the maximum), and relatively low approximation errors (note that the reference energy values are in the order 1E-13).

To also visually asses the approximation quality, we plotted for the two considered cases (i.e., (M,N) = (100,20), and (M,N) = (1000,20)) in Figure 2-7 and Figure 2-8 the fit line between the approximated energy values and the reference energy values, defined by the slope and y-intercept tabulated in Table 2-7, and a scatterplot of the reference energy values. In agreement with the MSE and MAE values, both plots reflect good fitting results.



Figure 2-7 Energy Approximation for (M, N) = (100, 20).



Figure 2-8 Energy Approximation for (M, N) = (1000, 20).

Table 2-8 and Table 2-9 summarize for the full NN configuration (8 input statistics and 8 outputs – output energy and 7 output statistics) the goodness of approximation figures. We obtained high correlation figures for both training and testing, while the MSE and MAE values for each of the output variable are listed in Table 2-9.

Parameter	Training		Testing (new samples)	
ranneter	Value	Value	Value	Value
# of samples	100	1000	100	1000
# of bits per sample	20	20	20	20
NN layers sizes	8:10:8	8:10:8	8:10:8	8:10:8
Epochs	204	416		
Slope	0.97	0.96		
Y-Intercept	0.02	0.031		
Correlation	0.98353	0.97991	0.98009	0.97747

Table 2-8 Training and Testing Energy and Output Statistics Approximation Results

Table 2-9 Training and	d Testing Energy and	Output Statistics	Approximation Error
------------------------	----------------------	-------------------	---------------------

Parameter	(M,N) = (100,20)		(M,N) = (1000,20)		-
Falameter	MSE	MAE	MSE	MAE	-
Energy	5.9273e-29	6.1528e-15	4.5462e-29	5.2421e-15	
PHo	0.0024	0.0403	0.0028	0.0423	
SWo	0.0082	0.0723	0.0060	0.0621	
$Pt00_o$	0.0072	0.0690	0.0074	0.0672	
$Pt01_o$	0.0063	0.0642	0.0077	0.0697	
	1				

$Pt10_o$	0.0075	0.0684	0.0078	0.0697
$Pt11_o$	0.0204	0.1180	0.0292	0.1363
$CC01_o$	0.0095	0.0793	0.0110	0.0842

Figure 2-9 and Figure 2-10 depict as in the previous single output NN case, the scatterplot of the approximated outputs, and the fit line between the approximated and the reference outputs, where each output point is 8-dimensional.



Figure 2-9 Energy and Output Statistics Approximation for (M, N) = (100, 20).



Figure 2-10 Energy and Output Statistics Approximation for (M, N) = (1000, 20).

One may note in the two figures that our approach provides reasonably good estimation accuracy. The estimation accuracy can be increased but only at the expense of the pre-characterization time (i.e., the NN training time), as there is no real impact on the actual evaluation time once the NN was trained.

Having pre-characterized each gate type in a circuit from the energy and output workload statistics standpoint as a function of the input workload statistics, the energy of a circuit can now be evaluated in a single pass for a given workload.

2.4. Conclusion

In this chapter we introduced a novel, composite, bottom-up, circuit-level energy assessment framework able to take into account the failure probability of individual comprising gates due to various fault-inducing factors (e.g., technology variability, environmental aggression). We addressed the following two cases: single transition based energy modeling and multiple-transitions based energy evaluation. Given that this framework is able to provide the means for accurate energy evaluation of circuits operating in faulty conditions it facilitates the performance assessment in terms of energy vs reliability of the different data storage, transport, and processing algorithms and architectures developed in the other i-RISC WPs.

3. Interconnects Energy Measurement

Abstract: Inter and intra-processor data and address buses omnipresent in modern computing platforms are rather long parallel interconnects (metal wires) placed adjacently with a very small distance in between. This geometry is exacerbating the crosstalk effects among neighbors, as well as increasing the power consumption and propagation delay, as a result of the crosstalk induced glitches. One possible approach to address the reliable data transport problem is to use information transformation techniques, e.g., coding. However, to allow the evaluation and comparison of different reliable data transport techniques, a simulation methodology able to evaluate interconnects performance in terms of, e.g., energy consumption, propagation delay) is required.

Publications: Unpublished work

3.1. Simulation Framework and Methodology

Subsequently we concern ourselves with both microstrips (for global interconnects, which are used for clock and signal distribution between the functional blocks, and supplying power/ground to all functions), and striplines (for the local and intermediate interconnects, which are employed within functional blocks). Cross-sectional views for 2-conductors microstrip and stripline are graphically illustrated in Figure 3-1 and Figure 3-2, respectively. We note that while microstrips ares only lower bounded by a dielectric layer striplines have dielectric layer both on top and bottom. We note that in both cases, an equivalent SPICE model needs to be constructed prior to carrying interconnects related simulations [Saini15].



Figure 3-1 Microstrip Sectional View.



A block scheme of the interconnect simulation framework for evaluating the performance of different architectures for energy effective and reliable data transport is depicted in Figure 3-3.





Figure 3-3 Interconnect Simulation Flow.

Figure 3-4 n Conductor Line Model [Synopsys].

The first step in the flow is to obtain the interconnect SPICE compatible model, for given specifications (e.g., wire length, number of parallel conductors), and technology parameters (e.g., related to the dielectric and metal layer stack - conductivity, dielectric permittivity, wire pitch, etc.). To this end, we employed the electro-magnetic field solver from Synopsys, Raphael. As for the current purpose interconnects are not meant to operate in the high frequency domain, we determine the RLGC parameters instead of the S parameters **[Synopsys]**. This implies that every wire is modeled as an RLGC equivalent component, as illustrated in Figure 3-4. Having determined the interconnect RLGC SPICE model, a SPICE simulation can be conducted to assess the interconnect performance under various scenarios. Figure 3-5 illustrates a simulation setup, which consists of input voltage sources for each wire, and the RLGC interconnect with both its transmitting and receiving ends buffered.



Figure 3-5 Simulation Setup.

3.2. Simulation Results

For the purpose of illustration, we present subsequently experimental data obtained by utilizing the aforementioned simulation framework, for two scenarios: an 8-wire data bus and a 9-wire data-bus. The SPICE simulation was performed using a commercial 45nm technology node, for nominal

operating conditions, and different wire lengths, i.e., 1mm, 2mm, 5mm, and 10mm, to cover local, medium, and long range interconnects. To allow for a fair comparison, the data transmission frequency was set as a function of the wire length. A dataset of 10000 bits per wire were transmitted over the bus, - more precisely 10000 random, uncoded bits per wire for the 8-wire bus, and 10000 coded bits per wire in the case of the 9-wire bus. As the purpose of the current section is an energy measurement simulation framework and methodology, details concerning the coding scheme for energy efficiency are omitted inhere. As quantitative indicators of the interconnect performance, the energy consumed was measured for both the 8-wire bus and the 9-wire bus, and the afferent data arrival profiles were analyzed. The energy values reflect both the static and the dynamic components, and are obtained by measuring the current supply voltage and integrating it over the entire transmission duration.





Figure 3-6 Wire Length vs. Energy Gain for the Coded 9-Wire Bus.



Figure 3-7 illustrates the energy consumed for the 8-wire bus (denoted as x8 in the figure), and 9wire bus (denoted as x9) for varying wire length. We note that the highest energy gain is achieved for longer length interconnects, as expected. Figure 3-6 details the coded 9-wire bus percentages of the energy consumed and of the energy gain, relative to the energy values of the uncoded 8-wire bus. The energy gain provides an indication on the upper bound of the energy budget available for the encoders/decoders circuitry. It can be observed that the energy gain for the coded 9-wire bus is above 50% for all considered wire lengths, which makes the coding scheme possibly promising.

When evaluating interconnects performance one may also be interested in the data profile arrival, which is desired to exhibit a relatively flat propagation delay trend. Otherwise stated, the bits on all wires are desired to arrive with as less variance as possible with respect to one another, in order to avoid additional transmission delay. For this purpose, the arrival time at interconnect receiving end was measured for each bit per wire.

In order to compare the distributions obtained for each bus per wire, a box-and-whiskers plot was employed, following the usual conventions, as illustrated in Figure 3-8 for convenience:

- The upper quantile marks the value above which 25% of the data points lay.
- In a similar manner, the lower quantile marks the data value below which lie 25% of the data points.
- The median value is the middle of the dataset (not necessarily the mean), which means that 50% of the data are greater than this value.
- The upper fence corresponds to the greatest value, excluding the outlier data points.

- The lower fence corresponds to the lowest value, excluding the outlier data points.
- The upper outlier points are those data points with a value greater than 3/2 times the upper quantile value.
- Similarly, the lower outlier points are those data points with a value lower than 3/2 times the lower quantile value.



Figure 3-8 Box-and Whisker Plot Convention

Figure 3-9 and Figure 3-10 illustrate the data arrival profile for the 8-wire bus and 9-wire bus, respectively. We observe that for the uncoded 8-wire bus, the bit arrival time for each wire exhibit a smaller spread when compared to the coded 9-wire bus, but the maximum arrival time is lower when compared to the maximum arrival time for the 8-wire bus. This has positive implications on the transmission clock frequency for the 9-wire bus, which can be increased for the 9-wire bus.



Figure 3-9 Data Arrival Profile for 8-Wire Bus



Figure 3-10 Data Arrival Profile for 9-Wire Bus

3.3. Conclusion

The performance simulation methodology introduced in this chapter allows us to accurately assess the energy consumption and data arrival profile corresponding to a given interconnect structure. Thus, it enables the comparison of several encoder-interconnect-decoder architectures targeting energy effective and reliable data transport under investigation in WP4.

4. Performance Enhancement of Multi-Level Fault Injection and Emulation

Abstract: In this chapter, we present two improvements for the fault injection and fault emulation methods and schemes developed in Deliverable D2.2 [i-RISC/D2.2]. The first one is represented by a three level reliability analysis methodology, which aims at evaluating the fault tolerance of the circuits described at Register Transfer Level (RTL) under probabilistic timing errors. The analysis is performed at three level of abstraction: Statistical Static Timing Analysis (SSTA) for standard cell components, estimation based on probability density function (PDF) propagation for characterization of combinational blocks, and simulated fault injection (SFI) performed at RTL. The proposed methodology has been used for reliability analysis of proof-of-concept LDPC decoders in Deliverable D6.1. Furthermore, in this chapter, we present a performance enhancement method for the FPGA based probabilistic fault emulation scheme developed and presented in Deliverable D2.2 [i-RISC/D2.2]. The proposed improvement targets FPGA performance increase, at expense of increased cost and lower accuracy. It is based on pseudo-random permutations within the shift register associated to the basic FPGA emulation scheme, while maintaining the number of active fault bits within the register.

Publications: Part of this work has been submitted in the following papers:

Ioana Mot, Oana Boncalo, and Alexandru Amaricai, "Performance Enhancement of Serial Based FPGA Probabilistic Fault Emulation Techniques", 18th International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS), Belgrade, 2015.

Alexandru Amaricai, Nicoleta Cucu-Laurenciu, Oana Boncalo, Jiaoyan Chen, Sergiu Nimara, Valentin Savin, and Sorin Cotofana, "Multi-Level Probabilistic Timing Error Reliability Analysis Using a Circuit Dependent Fault Map Generation", 30th Conference on Design of Circuits and Integrated Systems (DCIS 2015), Lisbon, 2015.

4.1. Multi-Level Probabilistic Timing Error Reliability Analysis

In this chapter, we propose a multi-level approach for analysis of complex systems described at RTL. The analysis is performed at three layers of abstraction:

- 1. SPICE simulations are employed for reliability characterization of standard cell components.
- 2. Analytical methods based on PDF propagation are used for gate level netlists.
- 3. Saboteur based SFI is performed for the reliability estimation of the RTL description

Regarding the first phase, Monte-Carlo SPICE based simulations are employed in order to perform SSTA for the standard cell components, under process, voltage and temperature (PVT) variations. An Inverse Gaussian (IG) PDF is obtained for each logic gates after the SSTA. The gate-level analysis is performed in an analytical manner, by applying a linear composition of the IG distribution's parameters obtained for the standard cell gates. The output of this phase is represented by the cumulative distributed function (CDF) of each Primary Output (PO) of the combinational blocks. The Cumulative Distributed Function (CDF) is used to determine the error probability for each PO for a given timing constraint. The same timing constraint, which represents the clock signal's period, is applied in the same clock domain. This way, we generate an accurate fault

map for the entire clock domain, which reflects the sensitivity of each PO to a given time constraint. The third phase is represented by a probabilistic SFI analysis of the RTL description.

We have employed the proposed methodology in order to analyze the error correction capability of the LDPC decoders analyzed in Deliverable D6.1 [i-RISC/D6.1]. In this chapter, we will present the proposed methodology, while the LDPC decoder analysis has been detailed in Deliverable D6.1 [i-RISC/D6.1].

4.1.1. Related Work

SFI has been widely used for evaluating the reliability of digital systems affected by different types of faults in early design phases **[Evans13] [SonzaReorda02] [Foutris14] [Jenn94]**. Development of the SFI methodologies is dependent on two goals: fault modeling capability and simulation overhead. Good fault modeling capability is obtained when using low-level circuit descriptions, such as gate level netlists. However, simulating complex systems at low abstraction layers is unfeasible. Furthermore, for probabilistic faults the large number of required simulations further aggravates the situation.

Several approaches have been developed which target the trade-off between the fault modeling capability and the simulation overhead **[Evans13][SonzaReorda02][Foutris14] [Hamad14]**. They rely on performing analysis on multiple layers of abstraction: fault models and fault behavior corresponding to higher abstraction layers are derived using analysis performed for low level descriptions of blocks, while the reliability of the entire system is estimated using high level analysis. The works in **[Evans13] [SonzaReorda02] [Hamad14]** propose methodologies to assess the reliability of digital systems described at RTL under Single Event Transient (SET) fault models. Static timing analysis for combinational blocks is used in **[SonzaReorda02]** in order to reduce the set of faults and to identify the faults, which may produce errors at blocks' primary outputs; these faults are then injected in the RTL model. The work in **[Hamad14]** uses SET fault injection for gate level characterization; the critical input combination and its probability is derived for combinational blocks; probabilistic model checking using PRISM is used for deriving the reliability at RTL. The approach in **[Evans13]**, uses SPICE simulation for characterizing SET effects at gate level; logic derating is used to determine the SET effects for combinational blocks; the results from this step are used for RTL SFI.

Our aim is to investigate the probabilistic timing errors, which mainly affect the CMOS circuits supplied at sub and near threshold voltages. Due to the process variations characteristic to deep subnanometer technologies, as well as supply voltage and temperature variations, sub-powered CMOS circuits exhibit a probabilistic behavior, with error probabilities dependent on the delay constraints. In order to accurately determine the probabilities for each fault location in the RTL description, we employ SSTA for deriving the PDF of each standard cell components; the obtained standard cell components' PDF are propagated in order to determine the CDF of each combinational block's PO. These CDF's represent the PO's error probability for a given delay.

4.1.2. Proposed Methodology

The proposed methodology is performed at three layers of abstractions:

- 1. Standard cell characterization SSTA based on Monte-Carlo SPICE simulation, is used in order to determine the propagation delay distribution for PVT variations for each standard cell component;
- 2. Gate level PDF propagation for each PO of each combinational block, the worst propagation path is determined; based on these paths, the delay distribution is derived using a linear composition of PDFs corresponding to standard cell gates on the critical path for each PO; the gate level netlists for each combinational block are obtained after the RTL design partitioning and logic synthesis, using the standard cell components characterized in the first phase;
- Saboteur based RTL SFI probabilistic saboteurs are inserted in the RTL description on each PO of the combinational blocks; the error probability for a given clock period is given by the PO's CDF;

Figure 4-1 depicts the reliability analysis flow. The proposed methodology makes use of commercial design and simulation tools, such as: Cadence Spectre or Virtuoso for standard cell characterization, Cadence Encounter RTL or Synopsys Design Compiler for logic synthesis, and Modelsim for RTL simulation.



Figure 4-1 – 3 Level Reliability Analysis Methodology

The first step in the reliability analysis flow depicted in Figure 4-1, consists of the statistical timing characterization of each standard cells of the technology library. Specifically, each standard cell is augmented with its propagation delay probability distribution over PVT variations. For each sampling set of process, supply voltage and temperature variation data, the cell propagation delay is derived as a mean between the measured rising and falling propagation delays which correspond to the two possible output switching situations, i.e., the output undergoing transition from logic "1" to logic "0", and vice-versa.

Monte-Carlo SPICE simulations are used in order to derive an Inverse Gaussian (IG) distribution for standard cell component. The IG based PDF, described in Deliverable D2.1 [i-RISC/D2.1] and [Chen14], has been used to model the probability delay characteristic of the standard cell components. This type of distribution is characterized by the mean μ and shape λ parameters.

The second phase of the proposed analysis is represented by the gate level analysis. It is used to derive the error probabilities for each PO of the combinational blocks. Using the IG based distribution of each standard cell component; the PDF of the PO is derived by employing a linear composition of the components on the worst delay path for that specific PO, as described in Deliverable D2.2.

The error probability of the PO for a given delay constraint is derived using the Cumulative Distributed Function (CDF) of the IG distribution. The CDF depicts the probability of the PO to be correct for a τ_{pL} timing constraint.

The last step is represented be the RTL SFI analysis, which is used to determine the reliability metrics for the analyzed circuit. The implemented SFI technique is based on the saboteur – HDL module which alters the value/timing characteristic of a signal [Jenn94]. The probabilistic saboteurs are applied at the inputs of sequential/memory components. The proposed saboteurs are composed of:

- 1. *Signal switch detection* this component is employed because timing errors manifest at output transitions
- 2. Random number generator (RNG) RNGs are used due to the probabilistic nature of the simulated timing errors
- 3. Logic XOR module this component is used for selective altering the "sabotaged" signal.

The saboteurs for each combinational output (or input for the sequential component) have as parameter an individual error probability. The error probabilities are given by the PO's CDF.

4.1.3. Case Study

The proposed methodology has been applied for the reliability analysis of data-path in the Min-Sum, Self-Corrected Min-Sum and FAID flooded LDPC decoders in Deliverable D6.1 [i-RISC/D6.1]. The three phases have been applied as follows:

- Standard cell characterization has been applied for NAND gates and memory elements
- Gate level PDF propagation has been applied to the pipeline stages within the variable node units and check node units of the three LDPC decoders, as well as the barrel shifters
- Saboteur based SFI has been applied at the RTL description of the LDPC decoders' architectures

The results of the performed reliability analysis have been detailed in Deliverable D6.1 [i-RISC/D6.1] and will not be presented here. However, it is worth mentioning that the proposed hierarchical approach allowed extracting LDPC decoder error correction capability measures - frame error rate, bit error rate and average number of iterations - after performing hundreds of thousands of simulations for each clock period value and each decoder.

4.2. Performance enhancement of serial based FPGA probabilistic fault emulation techniques

This chapter proposes an FPGA fault emulation technique, which improves the performance of the serial emulation scheme proposed in Deliverable D2.2 [i-RISC/D2.2]. It is based on pseudo-randomly permuting the fault bits within the shift register. It uses TRNG for fault bits generation and a modified shift-register for fault insertion and permutation. As in [i-RISC/D2.2][Boncalo14], a fault insertion phase is performed. It requires a number of clock cycles equal to the size of the shift register. However, after the fault insertion phase, multiple emulation clock cycles are performed, instead of 1. During these emulation cycles, the fault bits inserted during the insertion phase are pseudo-randomly permutated within parts of the shift-registers. This way, increased performance is obtained, on the expense of fault modeling accuracy – during an emulation cycle the faults are not uncorrelated - and higher cost – extra logic within the shift-register for permutations is required.

4.2.1. Serial FPGA emulation scheme

We have proposed a serial FPGA fault emulation scheme for probabilistic faults, which has been detailed in Deliverable D2.2 [i-RISC/D2.2]. The proposed scheme consists of a fault generator and control module, an autonomous testbench, and the corresponding observation logic. Regarding the fault generation and control, it is based on a True Random Number Generator (TRNG), which has been implemented using a Xilinx based design [Baetoniu08], and a shift register. The fault emulation consists of a fault insertion phase and a fault emulation phase. In the fault insertion phase, one fault bit is generated each clock cycle and is inserted into the serial shift register. After loading the shift register, one emulation cycle is performed. With respect to other emulation schemes, such as [May12][May13], it presents two advantages: low cost and high fault modeling capability (due to the uncorrelated nature of generated faults). However, it has lower emulation performance, due to the serial loading of the shift register.

4.2.2. Performance enhancement of the serial FPGA emulation scheme

The main performance penalty in the serial based fault emulation scheme is represented by the fault insertion phase required before each fault emulation clock cycle. We improve the performance by eliminating several fault insertion phases. This is achieved by applying pseudo-random permutations on the shift-register used for fault insertion. The proposed fault emulation is performed as follows:

- 1. Fault insertion the shift-register is loaded with the TRNG generated fault bits; this phase is identical to the one used in the serial based emulation scheme
- 2. Emulation phase -n emulation clock cycles are performed; the fault bits within the shift register are pseudo-randomly permutated each clock cycle.



Figure 4-2 – Performance Improved Serial FPGA Fault Emulation Scheme

Using this approach, a performance improvement of up to *n* times is obtained with respect to the approach in **[Boncalo14]**. Modified shift-registers, which perform the pseudo-random permutations are used. In order to preserve the probabilities obtained for the fault bits, after each random permutation the number of 1's within the shift register remains the same (e.g. a "100000" vector will permute in "000010" or "010000" or "000001" or "001000" or "000100"). The new fault emulation scheme is presented in Figure 4-2.

The increase in performance is achieved at the cost of higher cost (due to extra logic required to perform the permutations) and loss in the fault modeling accuracy. The latter is due to the fact that, except the emulation clock cycle after the fault insertion phase, correlations between faults do exist. In order to reduce the cost of the modified shift-register, pseudo-random permutations are performed on groups of 6-bits. Thus, the initial shift-register is replaced with a module consisting of 6-bit modified shift-registers. One reason to use blocks of 6 bits is represented by the modern FPGA structure, for which each logic element is composed of 6-input LUT and a D flip-flop. Each 6-bit block within the fault insertion module performs different types of permutations (e.g.one block performs the permutation from "001100" to "101000" while other block perform the permutation from "001100").

4.2.3. Evaluation

We have performed emulated fault injection on a 7-level 96 bits pipelined barrel shifter. This circuit is used for routing messages in LDPC decoders. The considered fault locations are the inputs for the pipelined registers. Therefore, the number of fault locations is equal to 672. For each level within the barrel shifter we have used one TRNG – shift-register module for fault injection. The fault emulation schemes have been implemented on a Xilinx Virtex-5 VLX-50T, speed grade -2 device. Xilinx ISE 14.7 has been used for synthesis and implementation.

	Cost	Overhead	
Modified Serial	3260 LUT-FF pairs	470%	
Woulled Serial	4 BRAM	47070	
Base Serial	2560 LUT-FF pairs	370%	
[Boncalo14]	4 BRAM	370%	
Barrel shifter	690 LUT-FF pairs	0%	

Table 4-1: Implementation Results for Fault Injected Barrel Shifter



Figure 4-3 – Insertion and emulation phase for the serial FPGA fault emulation scheme (a) and modified serial FPGA fault emulation scheme (b)

Table 4-1 indicates the cost of the fault emulation schemes applied to these circuits. We observe that the proposed one has a cost increase of 27% with respect to the serial based fault emulation scheme proposed in **[Boncalo14]**. The overhead is around 370% for the full serial emulation scheme and 470% for the proposed approach.

Regarding performance, Figure 4-3 depicts the difference between the serial based fault emulation scheme proposed in **[Boncalo14]** (a) and the proposed one (b) for 10 fault locations. For the analyzed barrel shifter, the serial based fault emulation scheme requires 96 clock cycles for fault insertion before one emulation clock cycles. The proposed solution uses 20 clock cycles of emulation before a new fault insertion phase (consisting of 96 clock cycles) is performed.

The faults are uncorrelated only in the first clock cycle of the emulation phase. In the following clock cycles of the emulation phase, the faults become correlated. This is due to the pseudo-random permutations within the shift registers. Therefore, the fault modeling capability is reduced.

The full parallel approach in **[May12]** using LFSR as RNG report an overhead of around 1000% for the ISCAS 89 s1196 benchmark circuit, for 18 fault locations, and around 8000% for flip-flops and 1000% for combinational LUT for the b14 benchmark circuit, for 220 fault locations.

4.3. Conclusion

In this chapter, we have presented the latest developments regarding fault injection based analysis of digital circuits under probabilistic errors. The first development is represented by a hierarchical methodology for simulation based reliability estimation of digital circuits described at RTL. It performs analysis at three abstraction layers: SSTA using Monte-Carlo SPICE simulation is performed in order to characterize the standard cell components, PDF propagation is employed at gate level, while saboteur based SFI is used to derive the reliability metrics at RTL. The proposed methodology outputs a fault map, which accurately reflects the timing violations errors due to overclocking across the entire circuit. The proposed methodology has been employed in order to derive the error correction capability of proof-of concept decoders developed for Deliverable 6.1 under probabilistic timing errors.

The second development represents a performance enhancement of the serial FPGA emulation scheme proposed in Deliverable D2.2. It employs pseudo-random permutation in shift registers. Therefore, with respect to the serial fault emulation scheme proposed in **[i-RISC/D2.2]**, it performs fewer fault insertions into the shift register; thus, a fault emulation campaign may take up to 18 times lower clock cycles with respect to serial base scheme. The disadvantages are represented by higher cost overhead, as well as lower fault modeling accuracy, as the faults are not anymore uncorrelated.

5. General Work Package 2 Conclusions

As this is the last WP2 deliverable we summarize in this final chapter our main achievements related to fault models and energy measures. Given that we carry on the discussion in a per objective fashion we first summarize the WP2 objectives as stated in the DOW:

- O2.1 Development of analytical failure models (statistical, Markovian, etc.) for both single and correlated errorsl;
- O2.2 Development of higher abstraction levels (gate level, RTL, functional) fault and error models and the corresponding simulated fault injection methodology;
- O2.3 Energy characterization of sub-powered CMOS gates in both static and transient regime.

The main contributions developed in this project associated to the O2.1 are:

- Linear compositional delay model based on an Inverse Gaussian (IG) Probability Density Function (PDF) – The developed mathematical delay model accurately capture the behavior of combinational circuits for both nominal power supply values, and in sub-powered regimes. Using the Cumulative Distribution Function (CDF) of the IG distribution provides accurate reliability estimates with respect to Monte-Carlo SPICE simulations, as well as good scalability.
- 2. IG distribution based timing analysis of sub-powered CMOS circuits We have proposed an accurate and comprehensive IG based delay model considering fan-out effects suitable for both combinational and sequential circuits. The proposed model provides both high accuracy with respect to Monte-Carlo SPICE simulation -, and great flexibility against process and voltage supply variations. The calculation of the IG distribution model key parameters is straightforward, which is helpful for large circuit delay estimation. With respect to Monte-Carlo simulations, the average error introduced by our approach is as low as 1.2% while, on the other hand, the simulation time is diminished by orders of magnitude.
- 3. Correlated error modeling and degradation quantification for PDF based circuit reliability assessment We proposed to employ a high-level degradation quantifier, based on an output voltage based PDF, in order to capture a gate (circuit) multiple correlated degradation effects, when being exposed to different aggression profiles. Furthermore, propagating such PDFs throughout a larger circuit the correlation between different comprised gates behavior is inherently captured, and thus the correlation of different errors encountered in the circuit is being accounted for.
- 4. Data dependent Markovian chain based symbolic analysis methodology Markov chains for both output data dependent error models, as well as input data dependent error models have been developed in order to evaluate the reliability of the error correction schemes built of unreliable components developed in WP4.

The main contributions developed in this project associated to the O2.2 are:

 Data dependent gate level Simulated Fault Injection (SFI) – We have proposed a mutant based SFI methodology, which implements four types of data dependent probabilistic errors. We have shown the flexibility of the proposed approach, as the probabilities associated to each logic gate in the netlist can be adjusted corresponding to each gates voltage and delay constraints.

- 2. Multi-level data dependent SFI for Register Transfer Level (RTL) descriptions We have provided a hierarchical approach in order to perform accurate data dependent SFI for RTL circuit descriptions. The RTL circuit is decomposed into simple blocks; data dependent gate level SFI is performed for these blocks; the results of these gate level simulations are used for deriving the SFI components at RTL level; RTL SFI is performed in order to estimate the reliability of the system.
- 3. **Multi-layer hierarchical reliability analysis for probabilistic timing errors** We have developed a three-layer reliability evaluation methodology for the analysis of the effects of probabilistic timing errors in digital CMOS circuits. It performs statistical static timing analysis for standard cell components, PDF propagation for gate level netlists and SFI for the estimation of the reliability of the RTL description.
- 4. Cost-effective FPGA probabilistic fault emulation We have developed a serial scheme for probabilistic fault emulation. The proposed scheme uses a true random number generator for fault bit generation and a shift register for fault bit insertion. The proposed scheme presents significant cost savings with respect to state of the art solutions, as well as more accurate fault modeling capability.

Regarding O2.3, the main contributions of this project are:

- 1. Hierarchical energy modeling of faulty CMOS circuits We have proposed a bottom-up energy assessment framework, which takes into account the failure probability of individual logic gates due to several fault-inducing factors. The proposed methodology has been proven both accurate and computational effective for the following two cases: (i) single transition based energy modeling and (ii) multiple-transitions based energy evaluation.
- Energy assessment framework for on-chip interconnects We have developed an energy measurement simulation based methodology, which relies on an electro-magnetic solver and on SPICE based simulations. The framework can be used to estimate both the energy measures and propagation delay for different interconnect configurations.

As a general conclusion of the proposed activities associated to Work Package 2 (WP2), we can state that we have made considerable progress with respect to the project's proposed objectives, as well as the state-of-the-art the reliability analysis of sub-powered probabilistic CMOS digital circuits. Moreover the main developments associated with WP2 have impacted the other technical WPs in the project, as the reliability and energy evaluation techniques and methodologies have been applied to evaluate the developed fault tolerant techniques, algorithms, and circuits.

References

[Boncalo14] O.Boncalo, A. Amaricai, C. Spagnol, E. Popovici "Cost effective FPGA probabilistic fault emulation", Proc. Nordic Microelectronics Event (NORCHIP), 2014

[Byrne15] C. L. Byrne, "Signal Processing: A Mathematical Approach", CRC Press, 2015

[Chen14] J. Chen, C. Spagnol, S. Grandhi, E. Popovici, S. Cotofana, and A. Amaricai, "Linear Compositional Delay Model for the Timing Analysis of Sub-Powered Combinational Circuits", Proceedings of the 2014 IEEE Annual Symposium on VLSI (ISVLSI), 2014

[Evans13] A. Evans, D. Alexandrescu, E. Costenaro, and L. Chen, "Hierarchical RTL-Based Combinatorial SER Estimation", Proceedings of the 19th International On-Line Testing Symposium (IOLTS), 2013

[Foutris14] N. Foutris, M. Kaliorakis, S. Tselonis, and D. Gizopoulos, "Versatile architecture-level fault injection framework for reliability evaluation: A first report", Proceedings of the 20th International On-Line Testing Symposium (IOLTS), 2014

[Hamad14] G. B. Hamad, O. Mohamed, and Y. Savaria, "Probabilistic model checking of single event transient propagation at RTL level", Proceedings of the 21st IEEE International Conference on Electronics, Circuits, and Systems (ICECS), 2014

[i-RISC/D2.1] FP7-ICT/FET-OPEN/ i-RISC project, Deliverable 2.1, "Circuit level fault models for sub-powered CMOS circuits for uncorrelated and correlated errors", February 2014

[i-RISC/D2.2] FP7-ICT/FET-OPEN/ i-RISC project, Deliverable 2.2, "Higher abstraction fault models and their simulation methodology", November 2014

[i-RISC/D6.1] FP7-ICT/FET-OPEN/ i-RISC project, Deliverable 6.1, "Report on Reliability Aware Synthesis and LDPC Decoders Built with Unreliable Components", April 2015

[Jenn94] E. Jenn, J. Arlat, M. Rimén, J. Ohlsson, and J. Karlsson, "Fault injection into VHDL models: the MEFISTO tool" Proceedings of the Fault Tolerant Computing Systems (FTCS), 1994

[May12] D. May, and W. Stechele, "An FPGA-based Probability-aware Fault Simulator", Proceedings of the International Conference on Embedded Computer Systems (SAMOS), 2012

[May13] D. May, and W. Stechele, "A resource-efficient probabilistic fault simulator "Proceedings of the 23rd International Conference on Field Programmable Logic and Applications (FPL), September, 2013

[Saini15] S. Saini, "Low Power Interconnect Design", Springer, 2015

[Seber03] G. A. F. Seber, and C. J. Wild, "Nonlinear Regression.", John Wiley and Sons, 2003.

[SonzaReorda02] M. Sonza Reorda, and M. Violante, "Fault List Compaction through Static Timing Analysis for Efficient Fault Injection Experiments" Proceedings of the 17th IEEE Symposium on Defect and Fault Tolerance in VLSI Systems (DFT), 2002

[Synposys] http://www.synopsys.com