

# Multi-Level Probabilistic Timing Error Reliability Analysis Using a Circuit Dependent Fault Map Generation

Alexandru Amaricai<sup>1</sup> Nicoleta Cucu-Laurenciu<sup>2</sup> Oana Boncalo<sup>1</sup> Joyan Chen<sup>2</sup> Sergiu Nimara<sup>1</sup> Valentin Savin<sup>3</sup>  
Sorin Cotofana<sup>2</sup>

<sup>1</sup> University Politehnica Timisoara, Romania, Email: {alexandru.amaricai, oana.boncalo}@cs.upt.ro

<sup>2</sup> Computer Engineering Department, TU Delft, the Netherlands, Email: {N.CucuLaurenciu, S.D.Cotofana}@tudelft.nl

<sup>3</sup> MINATEC Campus, CEA-LETI, Grenoble, France, Email : valentin.savin@cea.fr

**Abstract**—This paper proposes a methodology for timing error analysis of RTL circuit descriptions. The evaluation has three components: (i) statistical static timing analysis (SSTA) for standard cell components (ii) estimation based on probability density function (PDF) propagation for characterization of combinational blocks, and (iii) simulated fault injection (SFI) performed at RTL. Reliability characterization of basic components is derived using SSTA; PDF propagation is used to accurately capture the probabilistic error profile of each primary output (PO) of combinational blocks; RTL saboteur based SFI is employed in order to assess the reliability of the whole circuit. The proposed methodology is applied for the fault tolerance analysis of a flooded Min-Sum (MS) LDPC decoder.

**Keywords**—Simulated Fault Injection, Timing Analysis, Probabilistic Errors, Timing Errors

## I. INTRODUCTION

One of the most radical approaches in tackling the increased power issues in today's digital circuits is represented by the aggressive voltage scaling till near and below the MOSFET threshold voltage [1]. However, the aggressive voltage scaling has the drawback of decreased reliability of the semiconductor devices. The reliability issues are further augmented by the process variations characteristic to deep nanometer CMOS technologies. One of the most important reliability problems of sub-powered devices is represented by the timing errors, which are due to the logic circuits' inability to switch for a given delay constraint [2]. In this context, the development of reliability assessment methods for circuits affected by timing errors becomes critical.

In this paper, 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 [2][3]. An Inverse Gaussian (IG) PDF is obtained for each logic gate 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 PO of the combinational blocks. The 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 a flooded MS LDPC decoder affected by probabilistic timing errors. The PO of each combinational block in the LDPC decoder has been injected with a failure probability given by applying the same timing constraint, which represents the clock period. This way, we can estimate the decoding performance of overclocked LDPC decoders.

The paper is organized as follows: Section II describes related approaches for multi-level reliability analysis; the proposed methodology is presented in Section III; the results for the reliability analysis using the proposed methodology of a flooded MS LDPC decoder are described in Section IV; the last section is dedicated to the concluding remarks.

## II. RELATED WORK

SFI has been widely used for evaluating the reliability of digital systems affected by different types of faults in early design phases [4][5][6][7][8]. 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 layers of abstraction is unfeasible.

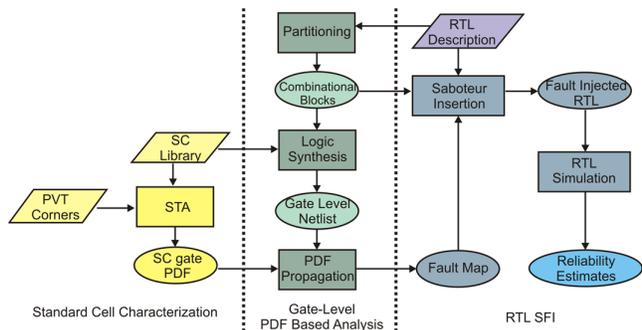


Figure 1 – Three level reliability analysis flow

Furthermore, for probabilistic faults, this problem is aggravated by the large number of required simulations.

Several approaches have been developed which target the trade-off between the fault modeling capability and the simulation overhead [4][5][6][7]. 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 [4][5][7] 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 [5] 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 [7] 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 [4], uses SPICE simulation for characterizing SET effects at gate level; logic de-rating 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 sub-nanometer 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.

### III. MULTI-LEVEL RELIABILITY ANALYSIS

#### A. Overview

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;
3. *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;

Fig. 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.

#### B. Statistical Static Timing Analysis

The first step in the reliability analysis flow depicted in Fig. 1, consists of the statistical timing characterization of each standard cell 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 IG distribution for standard cell component. The IG based PDF, described in [2], has been used to model the probability delay characteristic of the standard cell components. This type of distribution is characterized by the mean  $\mu$  and shape  $\lambda$  parameters.

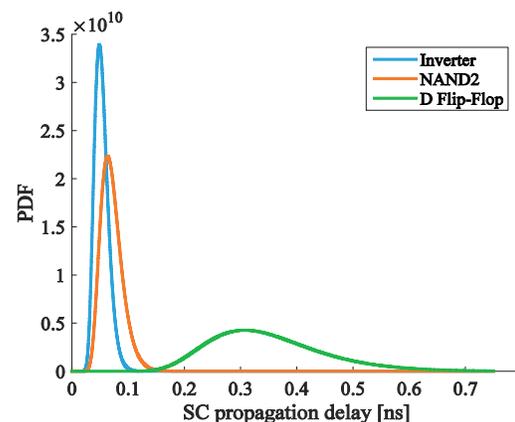


Figure 2 – Standard cell components delay IG based PDF

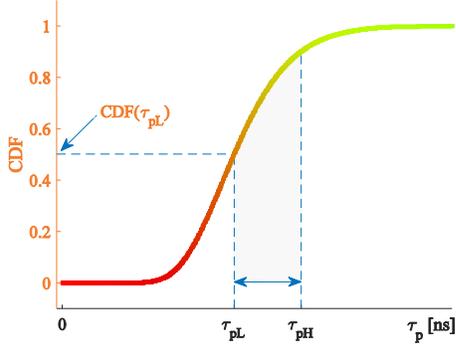


Figure 3 – Primary Output Propagation Delay IG Based CDF

The IG based PDF has the following advantages [2]: (i) it has been proved to provide accurate fitting with respect to the Monte-Carlo simulation results for both sub-powered and nominal supply-voltages (ii) PDFs of a combinational network can be easily derived using linear composition of the IG distribution of gates which compose it.

Fig. 2 depicts the IG based distribution for Inverter, 2-input NAND and D flip-flop implemented using PTM 45 nm MOS models, supplied at 0.35 V.

### C. Gate Level PDF Propagation

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. For instance, for a critical path being composed of 2 INV cells and 3 NAND2 cells, the mean  $\mu_o$  and shape  $\lambda_o$  parameters of the IG output propagation delay distribution are approximated as:

$$\begin{cases} \mu_o = 2 \cdot \mu_{INV} + 3 \cdot \mu_{NAND2} \\ \lambda_o = 2 \cdot \lambda_{INV} + 3 \cdot \lambda_{NAND2} \end{cases} \quad (1)$$

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  $\tau_{pL}$  timing constraint. Fig. 3 depicts the PDF (a) and the associated CDF (b) for an IG based distribution.

### D. Fault Map Generation

The PDF propagation via the means of linear composition of the standard cell components' IG distribution is used to derive the error probabilities for each PO. Fig. 4 depicts the CDF for a combinational block with three POs (denoted as  $O_1, O_2, O_3$ ), with different propagation delays.  $t_{clk}$  is the timing constraint, which is represented by the clock period. For a given  $t_{clk}$ , each PO will have its distinct error probability, given by its own CDF.

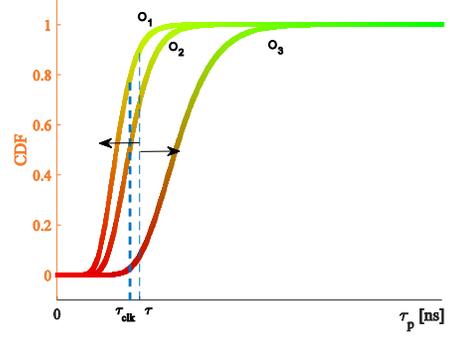


Figure 4 – Circuit POs Delay CDF

Regarding the error probabilities for different combinational blocks which operate in the same clock domain they are obtained from their respective POs' CDF, by applying the same timing constraint. Thus, combinational components with lower latencies will have lower error probability rates with respect to circuits with higher delays. Therefore, we are able to provide an accurate fault map over the entire circuit which to be used for RTL SFI.

### E. Saboteur Based RTL SFI

The implemented SFI technique is based on the saboteur – HDL module which alters the value/timing characteristic of a signal [8]. 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.

## IV. CASE STUDY – MIN-SUM LDPC DECODER

### A. Circuit under test

We have applied the proposed methodology for the fault tolerant analysis of a Min-Sum (MS) Low Density Parity Code (LDPC) decoder. The developed LDPC decoder implemented a flooded scheduling [10]. Flooded MS decoding is performed by exchanging messages between processing units, denoted as check node units (CNU) and variable node units (VNU). This message passing between the processing components is performed for several iterations, until a codeword is found or the maximum number of iterations is reached. We have implemented a MS decoder with serial processing for both variable node messages (denoted as  $\alpha$ ) and check node messages (denoted as  $\beta$ ). The input of the decoder is represented by the input log-likelihood ratios (LLR) – denoted as  $\gamma$  -, and outputs the hard decision bits, which are the signs of the a-posteriori LLR – denoted as  $\tilde{\gamma}$ .

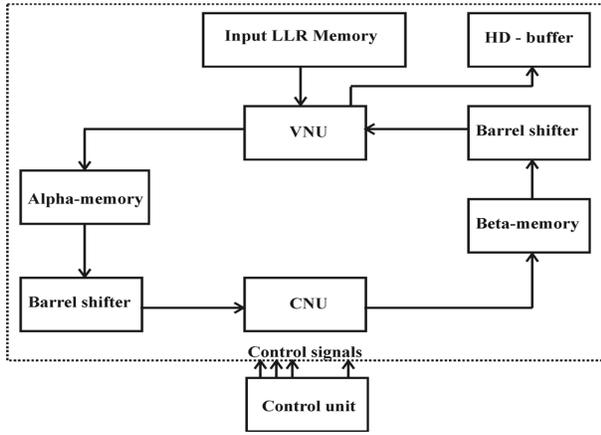


Figure 5 – Flooded MS LDPC decoder architecture

We have implemented the decoder for a (3,6)-regular LDPC code, with code length 1296 and code ratio  $\frac{1}{2}$ . The (3,6)-regular LDPC code is a quasi-cyclic LDPC (QC-LDPC) code [9], with the circulant matrix size of 54. The parity check matrix associated to this code contains 1296 columns and 648 rows, with the base matrix containing 24 columns and 12 rows.

The architecture of the implemented decoder is presented in Fig. 5. It consists of:

1. *Input Log Likelihood Ratio (LLR) memory* – this memory stores the input messages; these will be used in the decoding process for VNU computations; the memory word size is equal to quantization of  $\gamma$  (4 bits) multiplied by circulant size (54); the depth of the memory is equal to 24 (number of columns in the base matrix).
2. *VNU processing block* – it contains 54 individual VNUs; the 54 VNUs compute the corresponding variable-to-check messages ( $\alpha$ ) for a column in the base matrix, as well as  $\tilde{\gamma}$
3.  *$\alpha$  message memory* – it stores the variable check messages, which will be used in the check node computations; the memory word size is equal to the quantization of the  $\alpha$  message (4 bits) multiplied by the circulant size (54 $\times$ 4 bits); the depth of this memory is equal to 72.
4.  *$\alpha$  message Barrel Shifter (BS)* – it represents the routing network between the VNU outputs to check node unit inputs; it has 6 multiplexer (MUX) levels and a number of 54 $\times$ 4 (circulant size multiplied by  $\alpha$  message quantization) MUXes per level.
5. *CNU processing block* – it contains 54 individual CNUs; the 54 units compute the corresponding  $\beta$  for a row in the base matrix; a compressed form of the all 6  $\beta$  messages corresponding to a row in the parity check matrix is used; the size of the compressed  $\beta$  message is 15 bits.
6.  *$\beta$  message memory* – it stores the  $\beta$  messages, which will be used in the VNU processing; the memory word size is equal to 54 $\times$ 15 (circulant size multiplied by compressed  $\beta$  message size), while the memory depth is equal to the number of rows in the base matrix (12).

7.  *$\beta$  message BS* – it represents the routing network between the VNU outputs to check node unit inputs; it has 6 multiplexer (MUX) levels and a number of 54 $\times$ 15 MUXes per level.
8. *Hard-decision memory* – this memory contains the hard decision bits obtained after each iteration; the output of the decoder is read from this memory
9. *Control unit and early termination circuit* – it has the role of providing the appropriate sequence of operations, according to the flooded MS decoding; it provides: (i) the read and write addresses for the 4 memories; (ii) the shift amounts for the 2 BS; (iii) the corresponding control signals for both processing units (VNUs and CNUs); the early termination circuit is used for the verification of the parity check equations; if a codeword is found, decoding is stopped; otherwise, the decoder will perform the maximum number of iterations.

The BS has a single pipeline stage, the VNU has 4 combinational stages, while the CNU has 3 pipeline stages. The data read outputs of the memories are latched.

The synthesis results for the implemented LDPC decoder, using Xilinx ISE 14.7 for Xilinx Virtex-7 VX485T device, indicate that it occupies 17290 LUT-FF pairs and 20 BRAM blocks.

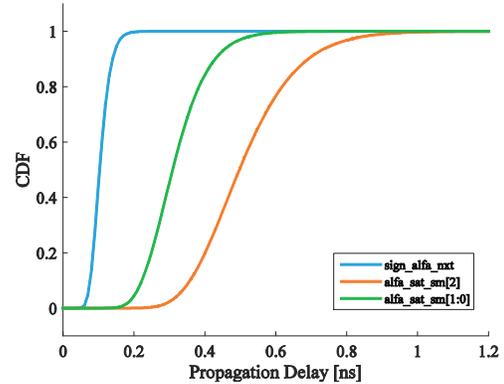


Figure 6 – CDF of CNU's first combinational stage

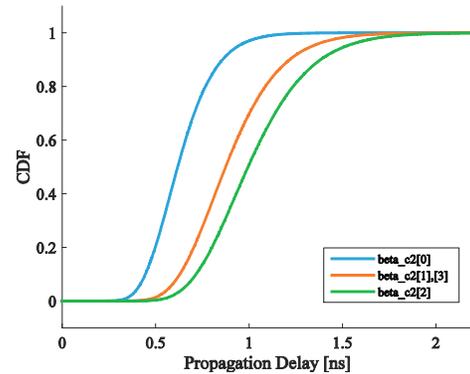


Figure 7 – CDF of VNU's first combinational stage

TABLE I – AVERAGE VALUES OF FAILURE PROBABILITIES FOR CONSIDERED COMPONENTS

Clock Period (ns)	Memory	BS	VNU	CNU
5.5	0.00E+00	0.00E+00	0.00E+00	1.28E-10
4.0	0.00E+00	0.00E+00	3.14E-08	3.51E-07
2.5	9.64E-07	2.79E-06	1.92E-04	6.05E-04
2.2	1.28E-05	2.71E-05	9.94E-04	2.33E-03
1.9	1.60E-04	2.62E-04	4.80E-03	8.13E-03

### B. Fault map generation

The combinational components have been synthesized with Cadence Encounter RTL tool, while the generated netlists are composed of Inverter and NAND gates. Regarding the analysis, we have injected faults only in the data-path computation block, without altering the control unit. This relates to the fact that injecting faults into the control unit can create severe disruptions in the LDPC decoder’s data flow, such as reading/writing messages from/to incorrect memory addresses or routing messages to the inappropriate processing units, which make the decoder unable to perform the LDPC decoding algorithms. Furthermore, the area associated to the control unit is less than 2% of the entire LDPC decoder. Thus, protecting the control unit by additional redundancy will not influence in a significant way the area of the decoder. Fig. 6 and 7 depict the CDF of the POs of the first CNU combinational stage and of the first VNU combinational stage.

The developed LDPC decoder has a single clock domain. Therefore, the same timing constraint is applied to all combinational blocks. Table I summarizes the average value average failure probabilities for the components used in the decoder, as derived for the CDFs of each combinational stage and memory block outputs, for different clock periods values. For clock periods higher than 5.5 ns, the failure probabilities are less than  $10^{-10}$ ; therefore, we have considered the decoder error free for these clock frequencies.

This fault map provides good overview of the error conditions under which the MS LDPC decoder will exhibit error correction capability degradation.

### C. Fault injection framework

The main goal of our analysis is to determine the error correction capability or decoding performance of the faulty MS LDPC flooded decoder. The considered metrics for the error decoding performance are: Frame Error Rate (FER) and Bit Error Rate (BER). The input messages for the decoder have been generate by C++ transmission channel model which consists of: (i) random word generator, which generates in a random manner the sequence of useful bits (ii) LDPC encoder – this component performs the encoding process using the word generate in the previous phase; it outputs a codeword (iii) channel model – it models the transmission channel and generates the corresponding noise, which alters the codeword. The output of the channel model represents the inputs of the RTL description of the LDPC decoder.

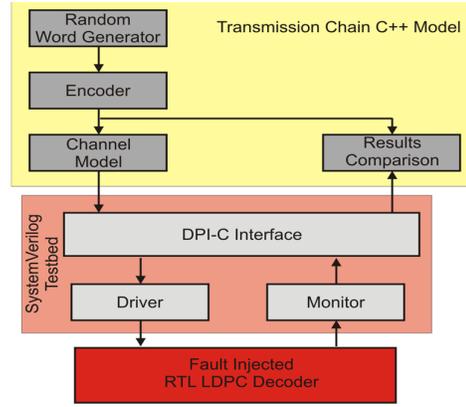


Figure 8 – SFI framework for LDPC decoder analysis

In order to determine the error correction capability of the LDPC decoder affected by timing errors, the outputs of the decoder have been compared with the codewords generated by the LDPC encoder used in the C++ transmission channel model. This represents a different strategy for reliability assessment of circuits, as it does not require the comparison of the outputs of the fault injected circuit with the outputs of the correct circuit (in this case, the output of the faulty LDPC decoder with the output of the correct decoder).

In order to perform this type of analysis, which outputs the decoding performance of the LDPC decoder, the following SFI framework based on SystemVerilog has been developed. It consists of (Fig.8):

- Transmission C++ model
- System Verilog wrapper and interface – this component extracts the input codewords from the transmission C++ models and feeds them to the RTL LDPC decoder
- RTL fault injected LDPC decoder

This type of SFI framework enables us to determine the error correction capability of the LDPC decoder for different channel parameters, such as the Signal-to-Noise Ratio (SNR) for Binary Additive White Gaussian Noise (BiAWGN).

### D. Simulation results

The simulations have been performed using Modelsim 10.2 commercial simulator. BiAWGN channel model has been employed. Simulations have been performed for an SNR range from 1 dB - 3 dB, with a 0.25 dB step. For an SNR value and a clock period, the simulations stop either when 100 erroneous frames (which could not be decoded) have been obtained, or when 200.000 frames have been simulated. We have performed simulation for the following clock periods: 5.5 ns, 3.1 ns, 2.5 ns, 2.2 ns, 1.9 ns.

Fig. 9 and Fig. 10 depict the FER and BER for the faulty LDPC decoders. These two figures indicate that the timing errors which appear in the MS LDPC decoder for clock periods of 3.1 ns and 5.5 ns do not lead to error decoding performance loss with respect to the fault free decoder.

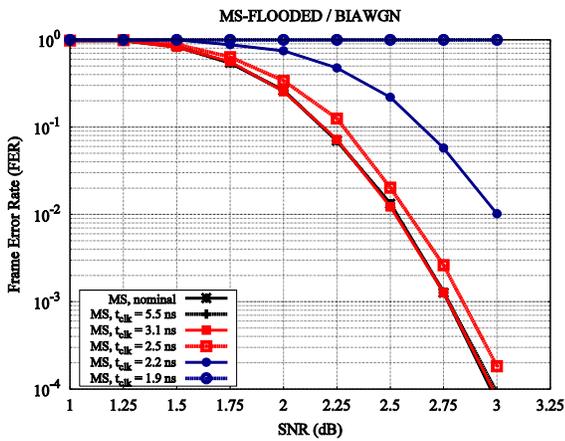


Figure 9 – FER for faulty MS LDPC decoder

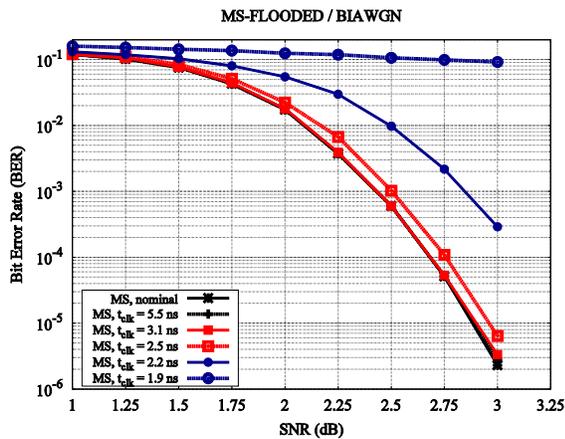


Figure 10 – BER for faulty MS LDPC decoder

A slight decoding performance degradation (of less than 0.1dB for a  $10^{-5}$  BER) is observed when clock frequency is increased to 400MHz (clock period of 2.5ns), while for clock periods of 2.2ns, we observe significant error correction capability loss. For a clock period of 1.9 ns, the MS LDPC decoder cannot decode anymore.

## V. CONCLUSIONS

This paper presents a three level reliability assessment methodology for probabilistic timing error analysis. The goal of the proposed analysis is to evaluate complex circuits' (described at RTL) behavior when subjected to timing violations. The proposed methodology combines SSTA performed in SPICE, gate level analytical techniques and saboteur based SFI performed at RTL. SSTA and PDF based propagation for gate level analysis has the role of determining

the failure probabilities for each combinational block's PO for a given clock frequency. This way, the obtained fault map reflects in an accurate way the timing violations due to overclocking across the entire circuit.

We have analyzed using the proposed methodology the error correction capability of an overclocked sub-powered flooded MS LDPC decoder. The analysis has been performed for BiAWGN channel model, from SNR range 1dB – 3 dB. The results indicate that increasing the clock frequency by a factor of 2 with respect to the maximum supported by the error free decoder will lead to no error correction capability degradation.

## ACKNOWLEDGMENT

This work has been supported by the European Commission Framework Program 7, project “i-Risc: *Innovative Reliable Chip Design from Low Power Unreliable Components*”, under Grant Agreement 309129.

## REFERENCES

- [1] H. Khaul, M. Anders, S. Hsu, A. Agarwal, R. Krishnamurthy, S. Bokhar “Near Threshold Voltage Design: Opportunities and Challenges” Proc. Design Automation Conference (DAC), 2012, pp. 1153-1158
- [2] J. Chen, C. Spagnol, S. Grandhi, E. Popovici, S. Cotofana, A. Amaricai “Linear Compositional Delay Model for the Timing Analysis of Sub-Powered Combinational Circuits” Proc. 2014 IEEE Annual Symp. On VLSI (ISVLSI), 2014
- [3] M. Merrett, P. Asenov, Y. Wang, M. Zwolinski, D. Reid, et al. “Modelling circuit performance variations due to statistical variability: Monte Carlo static timing analysis” Proc. 2011 Design Automation and Test in Europe (DATE), 2011
- [4] A. Evans, D. Alexandrescu, E. Costenaro, L. Chen “Hierarchical RTL-Based Combinatorial SER Estimation”, Proc. 19<sup>th</sup> Int. On-Line Testing Symp. (IOLTS), 2013
- [5] M. Sonza Reorda, M. Violante “Fault List Compaction through Static Timing Analysis for Efficient Fault Injection Experiments” Proc. 17<sup>th</sup> IEEE Symp on Defect and Fault Tolerance in VLSI Systems (DFT), 2002
- [6] N. Foutris, M. Kaliorakis, S. Tselonis, D. Gizopoulos “Versatile architecture-level fault injection framework for reliability evaluation: A first report” Proc. 20<sup>th</sup> Int. On-Line Testing Symp. (IOLTS), 2014
- [7] G. B. Hamad, O. Mohamed, Y. Savaria “Probabilistic model checking of single event transient propagation at RTL level” Proc. 21st IEEE International Conference on Electronics, Circuits and Systems (ICECS), 2014
- [8] E. Jenn, J. Arlat, M. Rimén, J. Ohlsson, J. Karlsson “Fault injection into VHDL models: the MEFISTO tool” Proc. Fault Tolerant Computing Systems (FTCS), 1994
- [9] M.P.C. Fossorier, “Quasicyclic Low-Density Parity-Check Codes from Circulant Permutation Matrices,” IEEE Trans. on Information Theory, vol. 50, no. 8, pp. 1788–1793, 2004.
- [10] F. R. Kschischang and B. J. Frey, “Iterative decoding of compound codes by probability propagation in graphical models,” IEEE Journal on Selected Areas in Communications, vol. 16, no. 2, pp. 219–230, 1998.