



Performance Analysis and Verification of Elastic Circuits Under Process Variations

Meysam Zaeemi,^a Siamak Mohammadi^{*a,b}^a School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran^b School of Computer Science, Institute of Fundamental Sciences (IPM), Tehran, Iran

PAPER INFO

Paper history:

Keywords:

Elastic circuits

Early evaluation

High-level modeling

Performance analysis

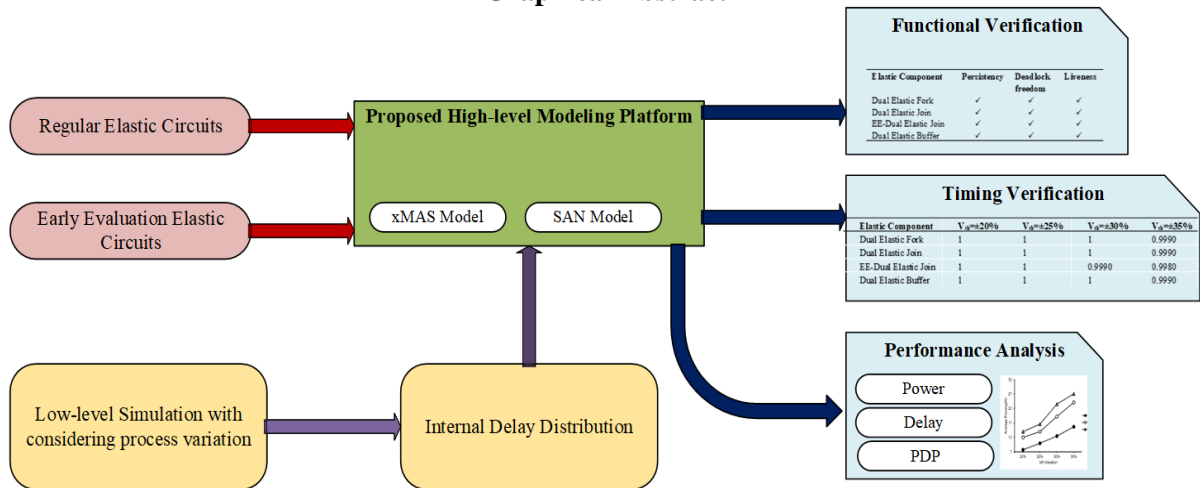
Process variation

A B S T R A C T

The diversity of designs and optimization methods for elastic systems makes evaluating their performance challenging. This paper proposes a method based on xMAS and high-level modeling to analyze the performance, functional and timing verification of regular and early evaluation synchronous elastic circuits while considering process variations. **The xMAS framework provides modularity, precise semantics, and executable models, enhancing formal verification and high-level analysis capabilities over existing approaches.** The proposed platform calculates the throughput value, which is the most critical performance factor in elastic circuits. The power, delay, and PDP of all early evaluation elastic components are evaluated under process variations and compared to those of regular elastic circuits. **The results indicate that early evaluation properties increase the sensitivity of circuit components to process variations, making their performance less predictable. Modeling results of the Elastic DLX microprocessor highlight these findings by demonstrating that process variations can cause a 26% reduction in throughput and lead to a 0.2% chance of synchronization errors between data and control signals. These findings underscore the critical need to account for process variations when designing and verifying early evaluation elastic circuits to maintain performance reliability.**

doi:

Graphical Abstract



1. INTRODUCTION

As technology advances and microchips become more sophisticated, synchronous circuits face challenges like clock skew and distribution. One potential solution to

these challenges is using asynchronous circuits. However, the adoption of this approach is hindered by the lack of mature design tools. An alternative middle-ground solution is the use of synchronous elastic circuits,

*Corresponding Author Email: smohamadi@ut.ac.ir (Siamak Mohammadi)

which can be designed using readily available commercial automated tools and are tolerant of different latencies. Elastic designs have been employed across various applications, showcasing remarkable efficiency. Within this domain, elastic structures have found utility in network-on-chip flow control (1-4), dataflow networks (5, 6), global asynchronous locally synchronous structures (7), convolutional neural network (CNN) accelerators (8), dynamic scheduling in high-level synthesis (9, 10), elastic silicon interconnects (11), and FPGA designs (12). Nevertheless, synchronous elastic circuits face their own challenges, including performance evaluation during early design stages for parameter selection, verification, and handling process variations.

A platform has been suggested (13) for high-level modeling of elastic circuits, which can also verify elastic circuits with process variation considerations. In addition to verification, there is a need for a rapid method to estimate the performance of elastic circuits. Such a method would enable architectural exploration in the early stages of elastic circuit design.

In (14-16), throughput values for elastic circuits have been calculated to evaluate their performances. Throughput values in synchronous elastic circuits are calculated via the number of valid transferred data in clock cycle units. Throughput in synchronous elastic circuits is measured by the amount of valid data transferred per clock cycle, making it a key metric for assessing circuit efficiency and functionality. High throughput indicates efficient data flow, essential for maximizing the performance of elastic systems that rely on continuous, dynamic data movement between components. Figure 1 shows control network of a simple synchronous elastic circuit. The illustrated circuit consists of two cycles, where the top cycle (blue dotted line) includes five buffers and four valid data (tokens). The bottom cycle (red dotted line) consists of four buffers and three tokens. To calculate the throughput of the circuit, throughputs for each cycle must be calculated, and the lowest value will be accounted for as the overall throughput of the system. In the top cycle, four tokens are transferred in five clocks, so the throughput is $4/5$, while the bottom cycle's throughput is $3/4$; therefore, the overall throughput is determined by the red dotted line cycle, which is $3/4$.

Other techniques, such as using early evaluation and insertion of variable latency units, have been presented to excel the performance of elastic circuits (17). However, the throughput evaluation method in those elastic circuits, which employ early evaluation elastic components is much more complicated, and the overall throughput is not equal to that of the slowest cycle. Early evaluation allows operations to proceed without waiting for all inputs, thereby potentially increasing data flow rates. However, this also introduces variability in processing times, especially under different operational

conditions, making the ability to accurately estimate and maximize throughput essential. Throughput directly reflects the circuit's capacity to handle data under varying conditions, thus serving as an indicator of robustness and efficiency in real-world scenarios where process variations may impact performance stability.

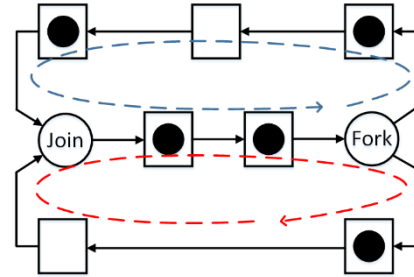


Figure 1. Control part of a synchronous elastic circuit

Several methods have been introduced to evaluate the performance of asynchronous systems. In (18, 19), a new simulation-based method has been developed, which performs statistical delay analysis on different asynchronous systems. Variability which has been a concern in these works, has been estimated based on mathematical methods for performance evaluation. In (20), a statistical framework is represented that is fit for low-level behavioral and performance analysis of synchronized pipelines in the presence of variations. Also, in (21, 22), branch folding and linear programming techniques are employed to improve the evaluation of performance bonds. In (23), Petri nets are used to evaluate the performance of an asynchronous pipeline in the presence of variations. Asynchronous and synchronous elastic circuits have structural differences, which means that these methods do not apply to synchronous elastic circuits. For example, the time between different transitions in asynchronous circuits, can be located in a continuous time phase (a few picosecond), while in elastic circuits, different transitions and latencies would be based on clock cycles.

The modeling implemented in (24) is one of the few works, executed on elastic pipelines. Petri net has been used to evaluate performance. In (25), a new elastic pipeline has been introduced, and the performance has been calculated analytically. In this work, no method was introduced to evaluate performance. Symbolic expressions were employed to introduce a method for performance evaluations of elastic circuits in (15). The exact value throughput was calculated in (14). This calculation was based on Markov chain analysis, although it has the same shortcomings as the exponential state explosion method. In all presented works, variations are ignored in performance evaluation of the synchronous elastic circuits. Additionally, none of the mentioned models can simultaneously perform verifications and performance analysis.

In this paper, we have added performance analysis abilities to the platform presented in (13). High-level modeling (using xMAS) and including process variations, specifically considering voltage threshold (V_{th}) variation as one of the crucial factors, leads to a precise performance evaluation of elastic circuits in early design stages. We have also modeled early evaluation components and related protocols. These components are capable of performance analysis and functional/timing verification.

The main contributions of this paper are as follows:

- Modeling all early evaluation components and relative protocols via xMAS.
- Performance analysis of all early evaluation components in terms of power, delay and Power-Delay-Product (PDP) under the influence of variations and comparing these results with those of regular elastic components.
- Adding performance analysis to the platform introduced in (13) was made possible by incorporating performance analysis capabilities into the early stages of the design process, thanks to high-level modeling.

The rest of this paper is organized as follows: Section 2 presents preliminaries for a better understanding of the prospective concepts of this paper, including elastic circuits and implemented optimizations such as early evaluation are introduced. Additionally, concepts such as process variations, xMAS and SAN models are introduced in this section. In Section 3, general flow of the proposed platform for performance analysis is introduced. Early evaluation components modeling and performance analysis techniques are described in this section, as well. Section 4 includes experimental results; all early evaluation components are tested for performance analysis, when process variations are present, and the results are compared with those of regular elastic components. All components have been verified in terms of timing and functionality. Finally, elastic DLX microprocessor has been modeled via the proposed method, and its timing/functional verification and performance analysis results are presented.

2. PRELIMINARY

2.1. Synchronous Elastic Circuits

Synchronous elastic circuits are based on synchronous and asynchronous circuits, which generally involve the advantages of both circuits. In addition to global clocks, signals are used to control the flow of data in this paradigm. Like asynchronous circuits, synchronous elastic circuits are tolerant against data latencies.

Elasticity in synchronous circuits is added thanks to elastic buffers and control networks. These networks consist of elastic components and handshaking signals.

These signals are in pairs of *Valid* and *Stop* signals. *Valid* signals are used to validate the data in channel. A component which cannot take a new signal, sends a *Stop* signal to the previous component to stall the flow of signals. Contrary to synchronous circuits, there is no need for the system to be in precise and synchronized clock cycles. In these circuits, the flow of valid data is checked and stored in each interface.

SELF (26) is the most well-known and the most efficient handshaking protocol for synchronous elastic circuits. With this protocol, three states are created using *Valid* and *Stop* signals.

- *Transfer* state ($Valid \wedge !Stop$): The sender prepares valid data and the receiver, receives it.
- *Idle* state ($!Valid$): The sender does not prepare a valid data.
- *Retry* mode ($Valid \wedge Stop$): the receiver does not receive the valid data prepared by the sender.

Elastic components that form elastic control network, consist of different types of *buffer*, *join* and *fork* components. Elastic *buffers* are similar to memory elements (flip-flops) in elastic circuits, which have the ability to store data and sending handshaking signals. Sending a *Stop* signal (from receiver to sender) indicates that the receiver has stopped working. It takes some time for the receiver to adapt with changes. Elastic *buffers* must prevent data loss. Internal structure of elastic *buffers* are introduced in (27).

Fork and *join* components are used for branch management in elastic control networks. In an elastic *join*, a valid output signal is asserted, when both input signals are valid. An elastic *fork* transfers *Valid* signals when the outlets are ready. Different types of *join* and *fork* components are studied in (13).

2.2. Early Evaluation Elastic Circuits

Tolerance to variations is one of the most important properties of synchronous elastic circuits due to their design, which leverages locally generated clocks and elastic handshaking control (27, 28). This property enables us to find a solution to increase the performance by altering the microarchitecture. Performance improvement in these circuits is achievable by using early evaluation.

In a functional unit, process starts when all inputs are ready. In some cases, these restrictions could be lifted; for example, in a multiplexer, select line chooses an input with one ready data. Thanks to early evaluation technique, there is no need to wait for the other input, in this case.

In early evaluation, an anti-token is placed on a channel that does not have a ready signal. This channel contains latent data, which will be later canceled out by anti-token.

Handshaking protocols and their components need to be extended to support early evaluation in elastic circuits (17). By adding early evaluation to the usual flow of tokens in control circuits, it is necessary to include anti-tokens in the backward flow.

In elastic circuits with early evaluation capabilities, two additional signals (other than *Valid* and *Stop* handshaking signals) exist to propagate anti-tokens backward. Token propagation signals are introduced as *Valid+* and *Stop+* in forward, while anti-token propagation signals in backward are introduced as *Valid-* and *Stop-*. Elastic buffers, elastic forks and elastic joins need to be extended to store and propagate anti-tokens (14).

A *dual join* controller includes a *join* controller to propagate tokens and a *fork* controller to transfer anti-tokens, in addition to various extra gates. Similarly, a *dual fork* controller consists of a *fork* controller to transfer tokens and a *join* controller to transfer anti-tokens.

Dual elastic buffer, *dual join*, and *dual fork* are responsible for the transfer of both tokens and anti-tokens. To enable early evaluation capabilities, a special *join* controller, namely “*early evaluation dual elastic join*”, is needed. Figure 2 shows the structure of these join controllers. The *early evaluation dual elastic join* controller comprises two main components: a *join* controller responsible for transmitting tokens forward (depicted in blue part) and a *fork* controller for transmitting anti-tokens backward (shown in gray part). For a more comprehensive understanding of these components, refer to (13). The presence of red gates in Figure 2 serves to prevent simultaneous assertion of the *Stop* bit and the *Valid* bit of the opposite polarity. Each output channel of the *fork* utilizes a flip-flop, which transitions to 1 in cases where the token fails to transmit to the output channel, indicating a *retry* state for that specific channel. Additionally, the yellow gate depicted in Figure 2 ensures that *Vout+* remains false if the *fork*, responsible for sending anti-tokens, has pending transfers. This mechanism prevents the propagation of tokens that require cancellation. The enabling function (EF) indicates the *join*’s function. For example, the *early evaluation dual elastic join* performs a function for a multiplexer control unit as follows:

$$EF = V_{sel}^+ \wedge ((sel \wedge V_a^+) \vee (\overline{sel \wedge V_b^+})) \quad (1)$$

in which *a* and *b* are multiplexer inputs, *sel* is the select line of multiplexer and *V_a*, *V_b* and *V_{sel}* are their *Valid* signals in forward direction. Gray parts have a structure, similar to *forks* that transfer anti-tokens. Green gates in Figure 2, generate the necessary anti-tokens.

2.3. Process Variation

Process variation occurs due to the variations in physical properties of the circuit (29). These variations are divided into two categories: systematic variations and non-systematic variations. Deterministic properties of systematic variations are introduced based on the variations in the electrical properties of two similar transistors with the same dimensions. Lithographic effects and chemical/mechanical polish can cause these systematic variations. Non-systematic variations are not deterministic, which means the outcome behavior of the system is unpredictable. Line edge roughness or random dopant fluctuations can cause non-systematic variations (30).

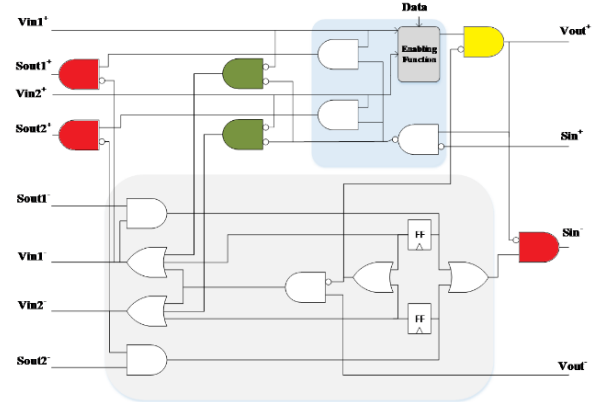


Figure 2. Structure of early evaluation dual elastic join

Variations in microchips fabrication process are inevitable, which affect the performance and reliability of the circuit. Due to the continuous changes in different components of the circuit, these variations bring up a bigger challenge in terms of performance evaluation of the system.

Manufacturing variations cause fluctuations in MOS parameters, which in turn cause variations in the performance of fabricated circuits. Gate’s width and length, oxide thickness and threshold voltage are the main sources of manufacturing variations.

To calculate and estimate variations before and during manufacturing process, statistic approaches are employed. Three classes of performance enhancement of circuits are used to improve the performance of the circuits, affected by manufacturing variations.

- Post-silicon calibration and repair: in this method, any shifts in circuit parameters are identified and corrected post-manufacturing by adjusting the *supply voltage*, *frequency*, *body bias*, or *clock skew* (31, 32).
- Variation avoidance: the aim of this approach is to detect failures in circuits during runtime and avoid

variations based on adaptive switching and source mappings (33).

- **Static timing analysis:** in this method, performance parameters of the system are statistically modeled. Circuits are designed to perform in the presence of variations. Static time variation evaluations are divided into deterministic STA and statistical STA, which will be reviewed in this paper.

Static timing analysis (STA) is classified into two categories: deterministic STA and statistical static timing analysis (SSTA) classes, where deterministic STA, is a more efficient method for timing verification. This method ensures timing constraints of the circuit after fabrication of the chip. Gates and interconnects latencies are deterministically evaluated. However, this method comes with its drawbacks. When a circuit is too large, it will be complicated to include all parameters.

In SSTA, variable sources are modeled as random variables with known probability distribution. An important approach in SSTA is to utilize Monte Carlo simulation (34, 35). This simulation offers a random process that is applied to a system, where it is hard to find a definition of its performance. To do this, varied parameters are generated randomly via probability distribution. Then, the design is simulated to find variation parameters. This simulation is performed for several times to calculate the performance distribution of the design. Monte Carlo simulation is precise but slow. Some techniques are introduced to improve Monte Carlo simulation (36, 37).

In addition to the mentioned methods, Finite Element Method (FEM) simulations, as presented in (38), are used for modeling axial and radial forces in amorphous core transformers, providing accurate insights into electromagnetic force variations. Although primarily applied for analyzing magnetic fields and forces, FEM can also aid in modeling distribution parameters in process variation, enabling precise predictions of its impact on circuit stability and performance. Moreover, in addition to Monte Carlo, neural network-based simulations, such as those presented in (39), can also be utilized to enhance analysis accuracy under process variations by leveraging advanced classification and prediction capabilities. Additionally, genetic algorithm-based simulations, as demonstrated in (40), could be applied to process variation analysis, offering an effective approach for optimizing parameters to improve stability and performance in variable environments.

Coefficient of variation (C_v) is used to evaluate the performance of circuits in the presence of variation based on Monte Carlo method, which is described as standard deviation over the average of the variable in the presence of variations, times the number of iterations of Monte

Carlo simulation. Coefficient of variation calculation method is as follows (41):

$$\mu(x) = \frac{\sum_{i=1}^n x_i}{n} \quad (2)$$

$$var(x) = \frac{\sum_{i=1}^n (x_i - \mu(x))^2}{n - 1} \quad (3)$$

$$\sigma(x) = \sqrt{var(x)} \quad (4)$$

$$C_v = \frac{\sigma(x)}{\mu(x)} \quad (5)$$

Where, μ is the average value for n simulations over x parameter. Equation 3 calculates the variance; deviation of different parameters over the average value, where its square root defines deviation σ (Equation 4).

2.4. xMAS Model

The xMAS framework has been developed by researchers at Intel corporations, which includes a compilation of basic primitives for high-level modeling of complex systems (42, 43). Figure 3 introduces these primitives by their symbols. Functions of each primitive are introduced here.

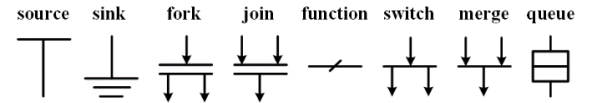


Figure 3. xMAS primitives

- **Source:** is a single-output primitive that enters data to the model non-deterministically.
- **Sink:** this primitive is the dual of *Source* that consumes data non-deterministically, with one input port.
- **Function:** this primitive model changes and transformations on inputs and produce outputs.
- **Queue:** is a standard-interface FIFO with two ports for reading and writing. This primitive has two parameters to indicate k as the number of data, and $type$ as the type of data.
- **Merge:** this primitive gives one output based on a variety of inputs. This primitive completes decision-making and multiplexing tasks.
- **Switch:** is a primitive to map data from an input to an output, with one input and two or more outputs to complete de-multiplexing task.
- **Fork:** Having one input and two outputs, this primitive takes inputs and gives them to both outputs. This only occurs when the input and both outputs are ready.
- **Join:** is the dual of *fork*, with two inputs and one output. In this primitive transfer occurs when all inputs and output are ready.

2.5. SAN Model

Stochastic Activity Networks (SANs) are used for modeling deterministic timing, statistical timing, and probability and distribution functions. These formalisms are stochastic extensions to Petri nets (44). Being able to complete various tasks, such as performance evaluation, performance verification, and reliability analysis, SANs consist of four primitives:

- *Places*: which holds a token that represents the system's state.
- *Activities*: include actions that take specific amount of time to complete. Instantaneous and timed (with deterministic or statistical timing based on the distribution function associated with its duration) are two types of activities
- *Input gates*: enabling of activities are controlled via *Input gates*, these gates also define the marking changes that will take place when an activity is complete. *Input gates* are defined via enabling predicates and functions.
- *Output gates*: define marking changes that take place when a function completes an activity.

Here, Mobius toolset (45) is used to simulate SAN models, with different iterations for simulations. These tools can define a variety of distribution functions, including Exponential, Normal, and Log-normal distribution functions. These tasks will lead to the analysis of the model using a generated trace file from each simulation.

3. PROPOSED WORKFLOW FOR PERFORMANCE ANALYSIS AND VERIFICATION

To verify synchronous elastic circuits and analyze their performance simultaneously, we have used tools and modeling methods introduced in Figure 4. Initially, all synchronous elastic circuits' components are modeled via xMAS. By putting them together using proper connections, an xMAS-based model of synchronous elastic circuit is created. Workcraft (46, 47) is used for this modeling. This modeled elastic circuit could also be functionally verified, using Workcraft. Three properties (*persistency*, *liveness* and *deadlock freedom*) are investigated at this stage.

Moreover, each elastic component is simulated using Monte Carlo method and HSPICE tool to extract delay distributions of each component in the presence of process variations. **Based on the findings of studies (48), V_{th} can have over twice the impact compared to L_n and over five times the impact compared to T_{ox} on power and PDP. This discrepancy becomes even more pronounced when considering their effects on delay, with V_{th} exerting approximately 10 times the influence of L_n and T_{ox} .**

Furthermore, the effect of W_{gate} variations is significantly lower than that of T_{ox} and L_{gate} (30).

Delay distribution extraction method for each elastic component is available in (13), in which V_{th} variation is considered as the most important factor in process variation. We have applied the same method in this paper.

The prepared model is translated to SAN model, using the built-in translator in xMAS model, and then, delay distributions extracted in the previous section, are added to the SAN model, which is completed via Mobius tool giving performance analysis and timing verification results.

3.1. xMAS Modeling of Elastic Early Evaluation Components

All of the components that could be applied in regular elastic circuits are modeled via xMAS and added to Workcraft library (13). For early evaluation however, elastic circuits' components must be extended. *Dual elastic buffer*, *dual elastic join*, *dual fork* and *early evaluation dual elastic join* that is used for early evaluation synchronous elastic circuits must be modeled with xMAS, and added to the library.

Figure 5 shows *dual elastic join*, modeled via xMAS. Different parts of the circuit are shown in different colors. The black part is a regular *join*, which transfers tokens. The Gray box is a regular *eager fork* (13), which transfers anti-tokens.

It is not possible to kill a token and stop it simultaneously in early evaluation of elastic circuits. This rule is indicated via the red part. The blue part testifies that no V_{out+} is asserted until the gray part is in the pending transfer mode.

Early evaluation dual elastic join is the most important component in elastic circuits with early evaluation capabilities that adds the ability of anti-tokens generation. Figure 6 represents *early evaluation dual elastic join* modeled with xMAS. This model is similar to *dual joins*, with two differences; the *enabling function* box, and the added blue part. While *dual joins* could only transfer anti-tokens, *early evaluation dual joins* can generate and transfer anti-tokens. The blue part in this model, completes anti-token generation task.

The *enabling function* box is modeled for different applications of *early evaluation dual elastic join*. As an example, the output of an *enabling function* box, used for a *multiplexer*, is represented in Equation 1, in which *sel* data is taken from data channel. Figure 7 represents an xMAS model implementing Equation 1. Other early evaluation components have been added to Workcraft library via the same method.

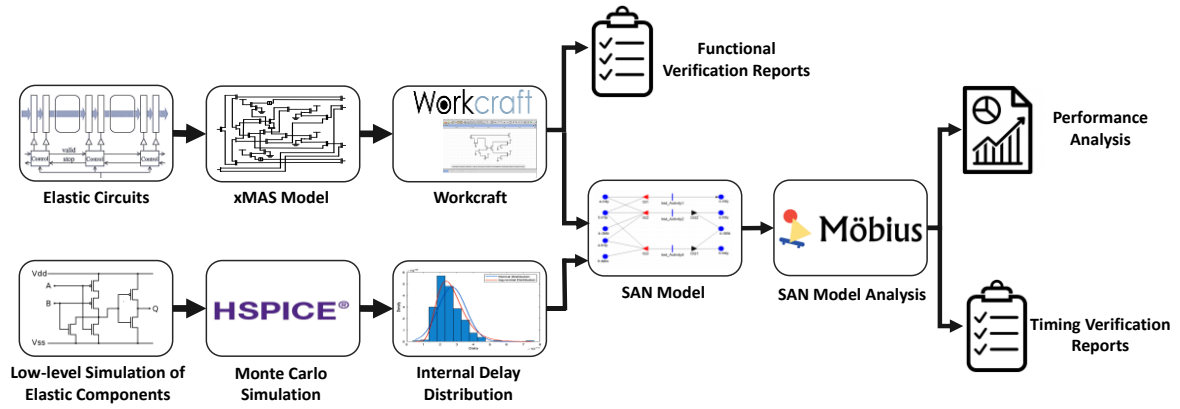


Figure 4. Performance analysis and verification workflow

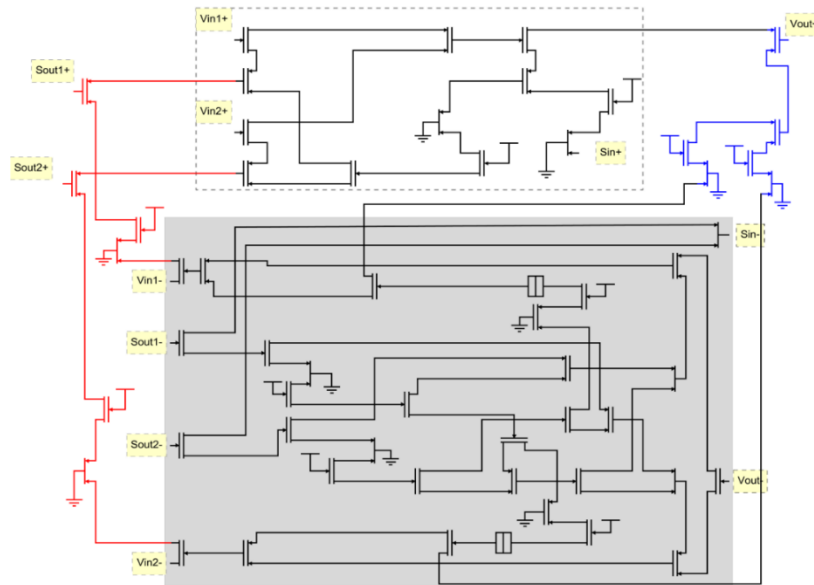


Figure 5. xMAS model of *dual elastic join*

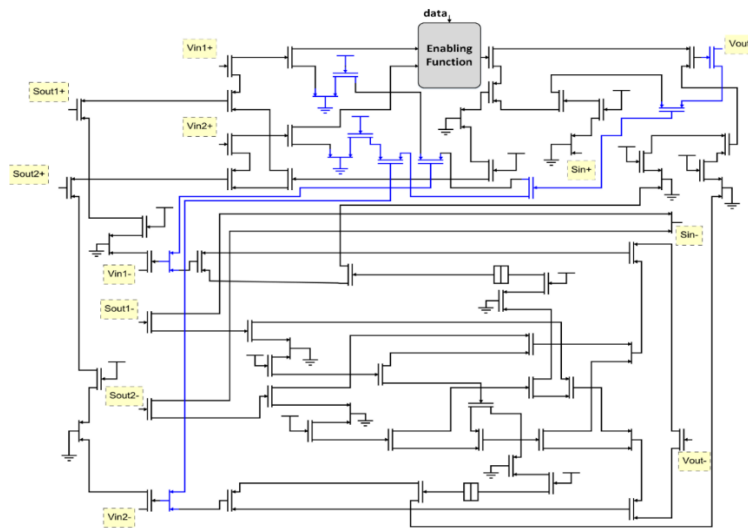


Figure 6. xMAS model of *early evaluation dual elastic join*

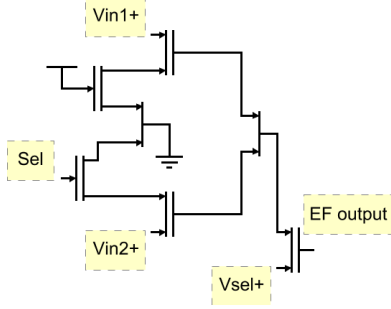


Figure 7. xMAS model of enabling function of Equation 1

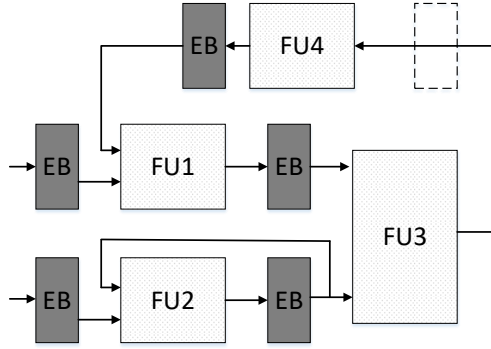


Figure 8. Sample of an elastic circuit

Table 1. Simulation parameters

Parameters	Values	Parameters	Values
Process	T-T	V_{th}	0.16 V
Temperature	25°C	T_{ox}	1 nm
Simulation Length	10 ns	L_{eff}	12.6 nm
Distribution	Gaussian	$L_n=L_p$	32 nm
V_{dd}	0.9 V	$W_n=W_p/2$	128 nm

3.2. Performance Analysis

It is possible to place desirable number of *buffers* in an elastic circuit (9). This property reduces the cycle period (*ns/cycle*), although it reduces the throughput (*token/cycle*). In Figure 8, an elastic circuit with four functional units and some *buffers* is represented. With a presumed *1ns* delay for each functional unit, an elastic *buffer* is placed between FU_3 and FU_4 to reduce the cycle period (floating dotted rectangular), which causes the reduction of cycle period in $FU_1 \rightarrow FU_3 \rightarrow FU_4 \rightarrow FU_1$ path from *2ns* to *1ns*. However, this does not mean a doubled-up performance, as this added *buffer* reduces the throughput from 1 to 2/3 in this path. By considering cycle period and throughput values, it could be deduced that token per time unit is increased from 1/2 to 2/3, which shows a 16.6% increase in performance. This result indicates that in order to make right decisions in designing elastic circuits, in addition to the cycle period,

we need to evaluate new throughput values of the new circuit, as well.

In this paper, we start designing a circuit using xMAS; then we will translate it to SAN model using the translator introduced in (13). By considering process variations and injecting extracted distribution delay functions into SAN model, the throughput evaluations using SAN model is completed. Moreover, by using probability consideration in Mobius, throughput values for elastic circuits with early evaluation capabilities are also calculable.

The proposed method for calculating throughput from the SAN model operates as follows: first, all possible cycles within the circuit are identified, and their sizes (i.e., the number of buffers in each cycle) are determined. Then, the throughput of each cycle is calculated independently of its repetition count. For circuits without early evaluation capability, the overall circuit throughput is taken as the minimum throughput among all cycles. In synchronous circuits with early evaluation, the repetition count of each cycle is computed using counters designated for each cycle, and the overall throughput is then obtained as the average throughput across cycles, weighted by their respective repetition counts.

4. EXPERIMENTAL RESULTS

4.1. The Study of Process Variations Effect on Early Evaluation Elastic Components

The control network of an elastic circuit consists of a variety of *fork*, *join* and *buffer* components, making the study of process variations and performance analysis essential. Process variations effect on the performance of *buffers*, *joins* and *forks* in regular elastic control networks have been studied in (13) and (49). Here, process variations effects on elastic components of early evaluation elastic control networks are studied and the results are compared with those of regular elastic networks' components.

Based on the findings of (30), the threshold voltage can vary up to 112%, as a peak-to-peak variation or as $\pm 56\%$ variation in V_{th} . We had investigated V_{th} variation for up to $\pm 35\%$ in this paper. These studies on early evaluation components' performance are completed for $\pm 20\%$, $\pm 25\%$, $\pm 30\%$, and $\pm 35\%$ variations for V_{th} , and the parameters in Table 1. Monte Carlo simulation is performed on each component with 256 iterations.

Average Delay, Power and PDP are represented in Figures 9 to 11, with V_{th} , as variation for *elastic fork* and *dual elastic fork* components. The results show an increase in variation to over $\pm 25\%$, causes a more

inclined increase in the power. Moreover, the *dual elastic fork* shows a poorer performance in comparison with regular *elastic fork* when the variation is set to $\pm 35\%$ of threshold voltage. The effect of variation (V_{th}) with respect to coefficient of variation (C_v) based on Equation 5 is represented in Figure 12. This figure shows that variations further affect the *dual elastic fork* (in comparison with *elastic fork*). It should be mentioned that the investigated *elastic fork* is the same as LF_{00} in (13).

Three types of *joins* have been investigated in Figures 13, 14, and 15, where the *dual elastic join* and *early evaluation dual elastic join* are used in early evaluation control networks. Increase in variations also increases power, delay and PDP. It is deduced that the average of power, delay and PDP of *early evaluation dual elastic join* components are higher than other *joins*, which is expected due to their complex circuitry. Early evaluation circuits are more sensitive to process variations than regular elastic circuits primarily due to their increased circuit complexity. These circuits include additional components, such as dual controllers for managing tokens and anti-tokens, and require specialized logic to handle partial data flows. This added circuitry introduces more gates and control paths, each of which can be affected by variations in manufacturing parameters like threshold voltage, gate length, and oxide thickness. Magnitude of delay and PDP for three types of *joins* is relatively similar for lower values of variation. The effect of variation on *early evaluation dual elastic join* is increased in higher magnitudes of variations (Figure 16). It is noteworthy that the investigated *early evaluation dual elastic join* refers to a 2:1 multiplexer.

Figures 17 to 20 represent the effect of V_{th} variation on *elastic buffer's* and *dual elastic buffer's* performance. The results indicate that the increase of variations, has a bigger effect on the *dual elastic buffer*. These effect reaches its peak at $\pm 35\%$ variations. It could be said that in comparison with regular elastic components, early evaluation components are more sensitive to variations, which also results in a bigger decrease in performance in early evaluation elastic circuits.

4.2. Functional Verification of Early Evaluation Elastic Components

Functionality of each early evaluation component is verified by formal methods in this section. To accomplish this, *persistence*, *deadlock freedom* and *liveness* of each component will be evaluated. Functional verification is performed by running xMAS via Workcraft and analyzing its trace files.

A component with no transitions from *Retry* to *Idle* in its channels, exhibits *persistence* properties. *Stop* and *Valid*

signals indicate the state in that elastic channel. *Retry* means *Valid* and *Stop* signals exist simultaneously, while *Idle* means no *Valid* signal is detected. Consequently, *persistence* property is investigated in each channel, based on $Retry \rightarrow Idle$ transition avoidance.

The existing properties of Workcraft are applied to evaluate *deadlock freedom* in channels. In order to study *liveness*, first we calculate the number of transferred tokens in each channel. These transferred tokens in different channels should match mathematically; this is available by evaluating result traces of workcraft run models.

Table 2. Functional verification of early evaluation components

Elastic Component	Persistence	Deadlock freedom	Liveness
Dual Elastic Fork	✓	✓	✓
Dual Elastic Join	✓	✓	✓
EE-Dual Elastic Join	✓	✓	✓
Dual Elastic Buffer	✓	✓	✓

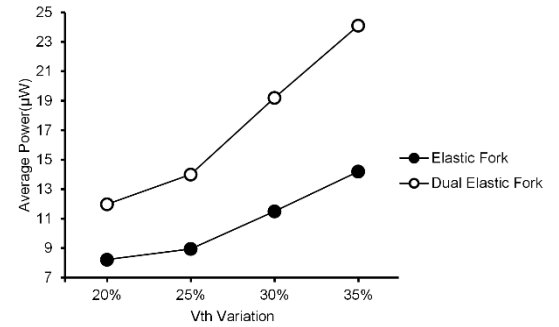


Figure 9. Average power of *elastic fork* and *dual elastic fork*

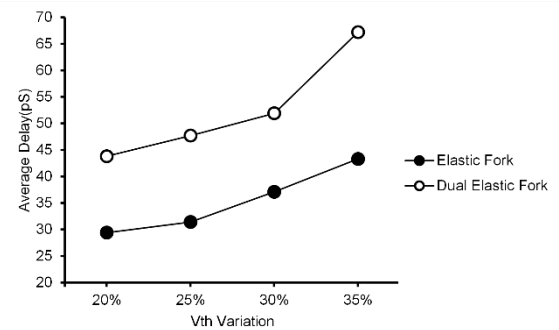


Figure 10. Average delay of *elastic fork* and *dual elastic fork*

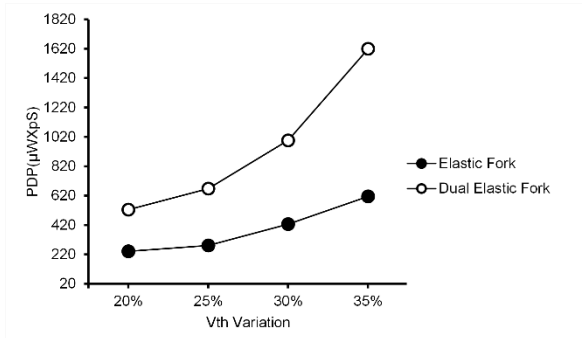


Figure 11. Average PDP of *elastic fork* and *dual elastic fork*

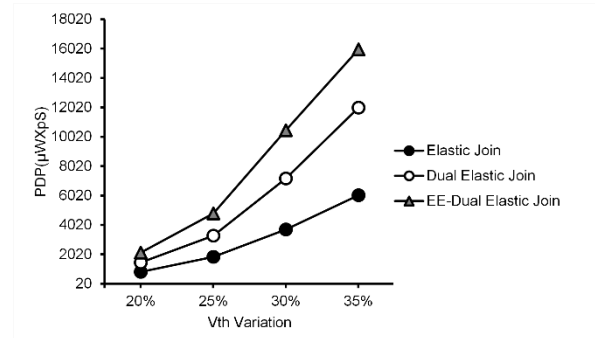


Figure 15. Average PDP of various *joins*

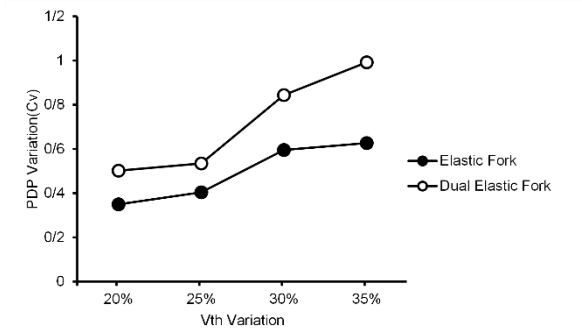


Figure 12. PDP variation effects on *elastic fork* and *dual elastic fork*

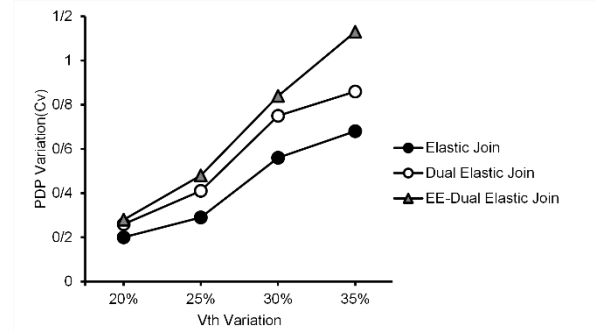


Figure 16. PDP variation effects on various *joins*

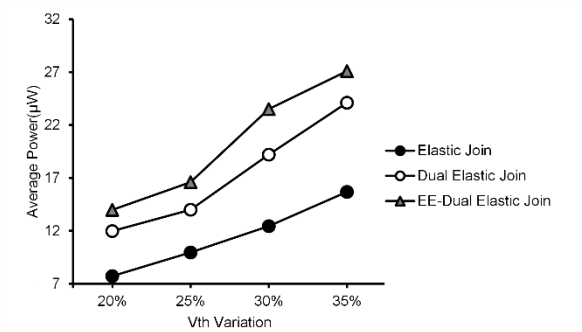


Figure 13. Average power of various *joins*

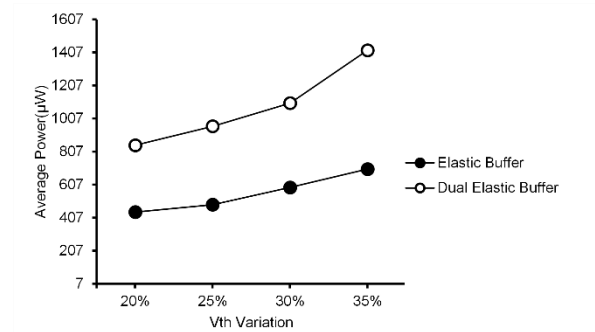


Figure 17. Average power of *elastic buffer* and *dual elastic buffer*

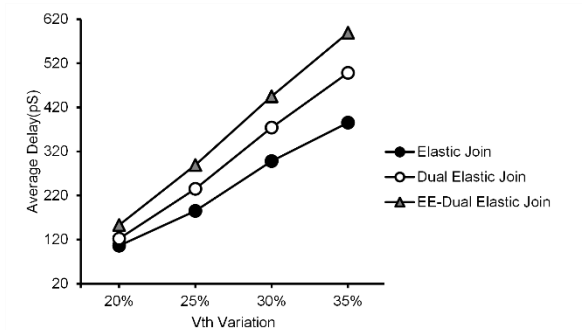


Figure 14. Average delay of various *forks*

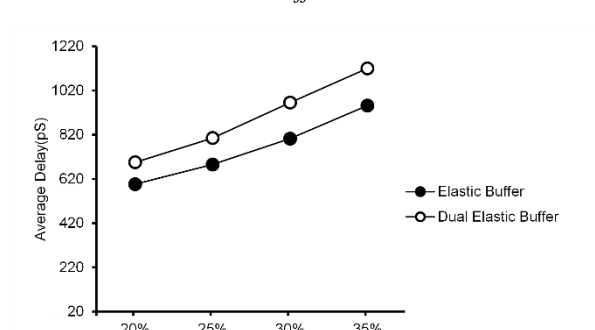


Figure 18. Average delay of *elastic buffer* and *dual elastic buffer*

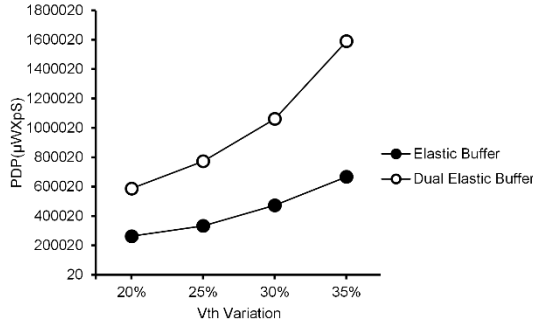


Figure 19. Average PDP of *elastic buffer* and *dual elastic buffer*

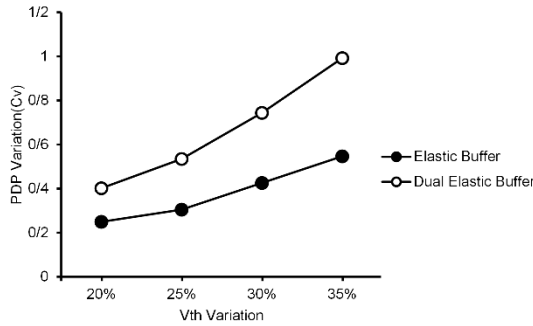


Figure 20. PDP variation effects on *elastic buffer* and *dual elastic buffer*

4.3. Timing Verification of Early Evaluation Elastic Components

For timing verification of early evaluation components, we have used the properties introduced in (13). By applying the property formulated in Equation 6, we will be able to investigate synchronization between data and control channels:

$$t_{valid(a-b)} + t_{enable} \geq t_{data(a-b)} + t_{setup} \quad (6)$$

Consider two *elastic buffers* (a and b) connected via $a-b$ path. In this equation, $t_{valid(a-b)}$ indicates *Valid* signal's transfer time in $a-b$ path. It is possible to place a number of *fork* and *join* components between these *buffers*. Similarly, $t_{data(a-b)}$ indicates data transfer time between the mentioned *buffers*. In this equation latch activation time and latch setup time are designated by t_{enable} and t_{setup} , respectively. Values of t_{enable} and t_{setup} are technology-dependent. Table 3 represents timing verification results for early evaluation components with V_{th} variation considerations.

Considered variation values for V_{th} are $\pm 20\%$, $\pm 25\%$, $\pm 30\%$ and $\pm 35\%$. To evaluate *dual elastic fork*, *dual elastic join* and *early evaluation dual elastic join*, each of these components are placed in $a-b$ control channel path. To evaluate *dual elastic buffer*, two *dual elastic buffer*

components are directly connected and simulated with 1024 iterations. Represented results in Table 3 indicate the correctness of Equation 6 over iterations (rate). These results indicate that for over $\pm 30\%$ V_{th} variation, all early evaluation components are prone to error, where *early evaluation dual elastic join* shows highest probabilities of malfunction (0.2%). In comparison with results in (13), it could be said that compared with regular elastic circuit components, early evaluation elastic circuit components are less prone to data channel and control channel synchronization error.

4.4. Verification and Performance Evaluation of Elastic DLX Microprocessor

Different versions of DLX microprocessor have been used to evaluate elastic circuits (14, 24, 50). The microprocessor introduced in (14) is used here to evaluate the method we proposed in this paper, which has early evaluation properties. **The DLX microprocessor was selected as a test case in this study due to its simple, well-understood architecture, which makes it an ideal model for analyzing elastic circuit performance. The DLX's modular design facilitates the evaluation of early evaluation techniques and process variation impacts across individual stages, which are crucial aspects of our performance analysis.** The elastic DLX microprocessor's control network is represented in Figure 21. Six *dual elastic buffers* are used in this elastic microprocessor, which is identical to EB_1 to EB_5 for five stages of pipeline (*Fetch*, *Decode*, *Execute*, *Memory* and *Write-back*) and to EB_6 , which is an empty *buffer* leading to a stall in the execution when bypassing the result to the next instruction is needed. Additionally, three *joins* and one *fork* are employed in this elastic architecture, of which two *joins* are *early evaluation dual elastic joins* (EEJ_1 and EEJ_2). Probability of data dependency between instructions is indicated by α . EEJ_1 receives data either from EB_6 with fixed probability of α or from the *decode* stage with probability of $1-\alpha$. EEJ_2 receives data either from *memory* stage before EB_5 with probability of β (which indicates the existence probability of load instruction), or directly from the execution stage with probability of $1-\beta$.

Functional verification results of the elastic DLX microprocessor show that this microprocessor satisfies all required properties (*deadlock freedom*, *liveness* and *persistency*). All possible paths between two *dual elastic buffers* in DLX microprocessors must be evaluated to complete the timing verification. These results are represented in Table 4. Error probabilities for synchronization between control and data channels over $EB_3 \rightarrow EB_4$ and $EB_3 \rightarrow EB_5$ paths are zero, while over other paths, error probability increases as V_{th} increases. $EB_2 \rightarrow EB_3$, $EB_4 \rightarrow EB_5$ and $EB_6 \rightarrow EB_3$ paths show highest

probabilities of error in our investigation, showing 0.2% probability of error for $V_{th}=\pm 35\%$.

Using Mobius properties, probable values for α and β for EEJ_1 and EEJ_2 components in SAN model are introduced, which help us calculate the throughput of the elastic DLX microprocessor. Some calculated throughput values are available in Table 5. To calculate each value, iteration is set to 1024 and the mean value is reported at the end. In the simulations, variations are set at $\pm 20\%$, $\pm 25\%$, $\pm 30\%$ and $\pm 35\%$. This result indicates that throughput of an elastic circuit decreases as the process

variations increase. This decrease in throughput is due to the reduction in the number of valid data in each cycle. The value of α immensely affects the throughput value, an increase in value of α , decreases throughput. That is because when α increases, the weight of $EB_3 \rightarrow EB_6 \rightarrow EB_3$ cycle increases in throughput calculations and since EB_6 in elastic DLX architecture does not have a token (is a *bubble*), it has the lowest throughput (0.5). For an elastic DLX architecture with no early evaluation properties, the slowest cycle determines the throughput of the circuit, which in this case is 0.5.

Table 3: Timing verification results for early evaluation components

Elastic Component	$V_{th}=\pm 20\%$	$V_{th}=\pm 25\%$	$V_{th}=\pm 30\%$	$V_{th}=\pm 35\%$
Dual Elastic Fork	1	1	1	0.9990
Dual Elastic Join	1	1	1	0.9990
EE-Dual Elastic Join	1	1	0.9990	0.9980
Dual Elastic Buffer	1	1	1	0.9990

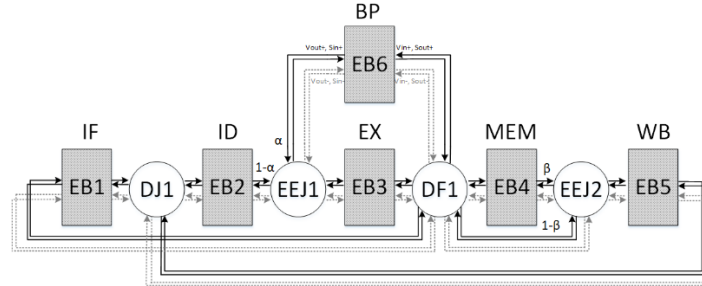


Figure 21. Elastic DLX microprocessor control network

Table 4. Timing verification results for for elastic DLX microprocessor

Path	$V_{th} = \pm 20\%$	$V_{th} = \pm 25\%$	$V_{th} = \pm 30\%$	$V_{th} = \pm 35\%$
$EB_1 \rightarrow EB_2$	1	1	1	0.9990
$EB_2 \rightarrow EB_3$	1	1	0.9990	0.9980
$EB_3 \rightarrow EB_6$	1	1	1	0.9990
$EB_3 \rightarrow EB_4$	1	1	1	1
$EB_3 \rightarrow EB_1$	1	1	1	0.9990
$EB_3 \rightarrow EB_5$	1	1	1	1
$EB_4 \rightarrow EB_5$	1	1	0.9990	0.9980
$EB_5 \rightarrow EB_2$	1	1	1	0.9990
$EB_6 \rightarrow EB_3$	1	1	0.9990	0.9980

Table 5. Throughput results for elastic DLX microprocessor

	$V_{th}=\pm 20\%$					$V_{th}=\pm 25\%$				
	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$
$\alpha = 0.1$	0.931	0.928	0.926	0.921	0.918	0.872	0.869	0.865	0.861	0.853
$\alpha = 0.2$	0.911	0.908	0.905	0.899	0.894	0.846	0.837	0.834	0.829	0.825
$\alpha = 0.3$	0.879	0.875	0.872	0.866	0.858	0.821	0.815	0.811	0.808	0.804
$\alpha = 0.4$	0.833	0.831	0.827	0.821	0.814	0.771	0.764	0.764	0.762	0.761
$\alpha = 0.5$	0.778	0.775	0.772	0.767	0.762	0.711	0.710	0.707	0.701	0.698
	$V_{th}=\pm 30\%$					$V_{th}=\pm 35\%$				
	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$
$\alpha = 0.1$	0.768	0.767	0.767	0.764	0.761	0.701	0.701	0.699	0.698	0.697
$\alpha = 0.2$	0.754	0.753	0.751	0.750	0.750	0.689	0.687	0.688	0.685	0.686
$\alpha = 0.3$	0.738	0.735	0.734	0.733	0.731	0.667	0.664	0.664	0.663	0.663
$\alpha = 0.4$	0.720	0.720	0.717	0.715	0.714	0.645	0.644	0.645	0.643	0.644
$\alpha = 0.5$	0.695	0.695	0.694	0.692	0.692	0.620	0.621	0.619	0.617	0.618

4.5. Model Validation

To the best of our knowledge, performance analysis and throughput measurement for early evaluation synchronous circuits by considering process variations have been completed in this paper for the first time. To validate our proposed model's accuracy, we have compared our results with those of (14). Their method calculates most precisely the throughput for early evaluation synchronous elastic circuits(24), however, they did not consider process variations in their work, and therefore, to be fair we have compared only our variation-independent results. Results of (14) are extracted from Markov chain analysis that suffers from exponential state explosion. Throughput calculations for DLX microprocessor have been completed with considerations introduced in (14), and the results are presented in Table 6. In our analysis, *memory* stage (similar to the one presented in (14)) is considered as a variable latency unit with $\delta+1$ as latency. Based on the obtained results, it could be deduced that throughput measurements based on our proposed model only differs by up to 3% from the results presented in (14). The difference between the results obtained in this paper and those presented in [14] is defined as "Error" in Table 6. Additionally, this model is compatible with high-level modeling which is used for complicated designs that help designers include process variations, and offer a work around to the exponential state explosion problem.

Table 6. Accuracy results of the proposed model

α	β	δ	Results of (14)	xMAS model results	Error
0.1	0.2	0.2	0.823	0.831	0.99%
0.1	0.2	0.4	0.714	0.725	1.48%
0.1	0.2	0.8	0.556	0.562	1.09%
0.1	0.3	0.2	0.82	0.812	-1.01%
0.1	0.3	0.4	0.714	0.727	1.77%
0.1	0.3	0.8	0.555	0.540	-2.77%
0.2	0.2	0.2	0.795	0.791	-0.50%
0.2	0.2	0.4	0.712	0.733	2.91%
0.2	0.2	0.8	0.556	0.568	2.06%
0.2	0.3	0.2	0.792	0.812	2.44%
0.2	0.3	0.4	0.711	0.730	2.63%
0.2	0.3	0.8	0.555	0.544	-1.94%
0.3	0.2	0.2	0.754	0.769	1.96%
0.3	0.2	0.4	0.705	0.725	2.72%
0.3	0.2	0.8	0.556	0.539	-3.09%
0.3	0.3	0.2	0.751	0.762	1.48%
0.3	0.3	0.4	0.702	0.719	2.34%
0.3	0.3	0.8	0.555	0.571	2.82%

4.6. Conclusion

This study presents a comprehensive analysis of the performance of synchronous elastic circuits with early evaluation under process variations. Our results show

that, with a V_{th} variation of $\pm 35\%$, throughput can decrease by up to 26%, highlighting the substantial impact of process variations on elastic circuits. This reduction underscores the need for careful consideration of process variability during design to maintain stable performance. Additionally, the early evaluation components in these circuits demonstrate higher sensitivity to variations compared to regular components, as they introduce more complex paths and timing requirements.

The proposed solution effectively addresses these challenges by using high-level xMAS modeling to predict performance and ensure functionality under variable conditions. Our framework enables early-stage performance optimization, helping designers implement effective adjustments to mitigate variation impacts on throughput and synchronization accuracy. Furthermore, the 0.2% probability of synchronization errors between data and control channels indicates areas where future designs can focus on robustness improvements.

While xMAS modeling offers a promising approach for high-level circuit design, it still lacks advanced tools for comprehensive statistical analysis and timing distribution evaluation. Currently, Workcraft is the most complete tool for xMAS modeling, but it has limitations. Future work could focus on enhancing xMAS modeling tools by integrating statistical analysis capabilities and probability distribution functions, which would significantly improve timing analysis and provide a more robust platform for circuit optimization under variability.

5. ACKNOWLEDGMENTS

This research was in part supported by a grant from the Institute for Research in Fundamental Sciences (IPM) (Grant No. CS1401-4-77).

6. REFERENCES

1. Yasudo R, Matsutani H, Koibuchi M, Amano H, Nakamura T. Scalable networks-on-chip with elastic links demarcated by decentralized routers. *IEEE Transactions on Computers*. 2016;66(4):702-16. 10.1109/TC.2016.2606597.
2. Michelogiannakis G, Dally WJ. Elastic Buffer Flow Control for On-Chip Networks. *IEEE Transactions on Computers*. 2013;62(2):295-309. 10.1109/TC.2011.237.
3. Seitanidis I, Psarras A, Kalligeros E, Nicopoulos C, Dimitrakopoulos G, editors. *ElastiNoC: A self-testable distributed VC-based network-on-chip architecture*. 2014 Eighth IEEE/ACM International Symposium on Networks-on-Chip (NoCS); 2014: IEEE. 10.1109/NOCS.2014.7008772.
4. Montano F, Ould-Bachir T, Mahseredjian J, David JP, editors. *A low-latency reconfigurable multistage interconnection network*. 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE); 2019: IEEE. 10.1109/CCECE.2019.8861540.

5. Edwards SA, Townsend R, Barker M, Kim MA. Compositional dataflow circuits. *ACM Transactions on Embedded Computing Systems (TECS)*. 2019;18(1):1-27. 10.1145/3274280.
6. Xu J, Josipović L, editors. Suppressing Spurious Dynamism of Dataflow Circuits via Latency and Occupancy Balancing. *Proceedings of the 2024 ACM/SIGDA International Symposium on Field Programmable Gate Arrays*; 2024. 10.1145/3626202.3637570.
7. Mamaghani MJ, Krstic M, Garside J, editors. Automatic clock: A promising approach toward GALSification. 2016 22nd IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC); 2016: IEEE. 10.1109/ASYNC.2016.20.
8. Seo Y, Lee S, Kim S, Wang J, Park S, Park CS, editors. Latency-insensitive controller for convolutional neural network accelerators. 2019 International SoC Design Conference (ISOC); 2019: IEEE. 10.1109/ISOC47750.2019.9027661.
9. Josipović L, Sheikha S, Guerrieri A, Ienne P, Cortadella J. Buffer placement and sizing for high-performance dataflow circuits. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*. 2021;15(1):1-32. 10.1145/3477053.
10. Josipovic L, Brisk P, Ienne P, editors. From C to elastic circuits. 2017 51st Asilomar Conference on Signals, Systems, and Computers; 2017: IEEE. 10.1109/ACSSC.2017.8335150.
11. Demme J. Elastic Silicon Interconnects: Abstracting Communication in Accelerator Design. *arXiv preprint arXiv:2111.06584*. 2021. 10.48550/arXiv.2111.06584.
12. Abbas M, Betz V, editors. Latency insensitive design styles for FPGAs. 2018 28th International Conference on Field Programmable Logic and Applications (FPL); 2018: IEEE. 10.1109/FPL.2018.00068.
13. Zaeemi M, Mohammadi S. High-level Modeling and Verification Platform for Elastic Circuits with Process Variation Considerations. *ACM Journal of Emerging Technologies in Computing System*. 2022. 10.1145/3534971.
14. Júlvez J, Cortadella J, Kishinevsky M, editors. Performance analysis of concurrent systems with early evaluation. *Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*; 2006. 10.1145/1233501.1233590.
15. Galceran-Oms M, Cortadella J, Kishinevsky M, editors. Symbolic performance analysis of elastic systems. 2010 IEEE/ACM International Conference on Computer-Aided Design (ICCAD); 2010: IEEE. 10.1109/ICCAD.2010.5653886.
16. Cortadella J, Petit J, editors. A hierarchical mathematical model for automatic pipelining and allocation using elastic systems. 2017 51st Asilomar Conference on Signals, Systems, and Computers; 2017: IEEE. 10.1109/ACSSC.2017.8335149.
17. Cortadella J, Kishinevsky M, editors. Synchronous elastic circuits with early evaluation and token counterflow. 2007 44th ACM/IEEE Design Automation Conference; 2007: IEEE. 10.1145/1278480.1278587.
18. Yahya E, Fesquet L, Ismail Y, Renaudin M, editors. Statistical static timing analysis of conditional asynchronous circuits using model-based simulation. 2013 IEEE 19th International Symposium on Asynchronous Circuits and Systems; 2013: IEEE. 10.1109/ASYNC.2013.12.
19. Yahya E, Renaudin M, editors. Performance modeling and analysis of asynchronous linear-pipeline with time variable delays. 2007 14th IEEE International Conference on Electronics, Circuits and Systems; 2007: IEEE. 10.1109/ICECS.2007.4511233.
20. Liu T-T, Rabaey JM, editors. Statistical analysis and optimization of asynchronous digital circuits. 2012 IEEE 18th International Symposium on Asynchronous Circuits and Systems; 2012: IEEE. 10.1109/ASYNC.2012.21.
21. Najibi M, Beerel PA, editors. Performance bounds of asynchronous circuits with mode-based conditional behavior. 2012 IEEE 18th International Symposium on Asynchronous Circuits and Systems; 2012: IEEE. 10.1109/ASYNC.2012.27.
22. Najibi M, Beerel PA, editors. Integrated fanout optimization and slack matching of asynchronous circuits. 2014 20th IEEE International Symposium on Asynchronous Circuits and Systems; 2014: IEEE. 10.1109/ASYNC.2014.17.
23. Mosaffa M, Mohammadi S, Safari S. Statistical analysis of asynchronous pipelines in presence of process variation using formal models. *Integration*. 2016;55:98-117. 10.1016/j.vlsi.2016.04.003
24. Galceran-Oms M. Automatic pipelining of elastic systems: PhD thesis, UNIVERSITAT POLITCNICA DE CATALUNYA; 2011.
25. Rezaei H, Moghaddam SA, Rahmati A. High-performance dynamic elastic pipelines. *Microprocessors and Microsystems*. 2018;56:113-20. 10.1016/j.micpro.2017.11.004.
26. Cortadella J, Kishinevsky M, Grundmann B, editors. SELF: Specification and design of synchronous elastic circuits. *International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems (TAU)*; 2006.
27. Carmona J, Cortadella J, Kishinevsky M, Taubin A. Elastic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2009;28(10):1437-55. 10.1109/TCAD.2009.2030436.
28. Ryu S, Koo J, Kim W, Kim Y, Kim J-J. Variation-tolerant elastic clock scheme for low-voltage operations. *IEEE Journal of Solid-State Circuits*. 2021;56(7):2245-55. 10.1109/JSSC.2020.3048881.
29. Howldar S, Balaji B, Srinivasa Rao K. Gate Oxide Thickness and Drain Current Variation of Dual Gate Tunnel Field Effect Transistor. *International Journal of Engineering*. 2024;37(3):520-8. 10.5829/ije.2024.37.03c.09.
30. Ezz-Eldin R, El-Moursy MA, Hamed HF. *Analysis and design of networks-on-chip under high process variation*: Springer; 2015. 10.1007/978-3-319-25766-2.
31. Tschanz JW, Narendra S, Nair R, De V. Effectiveness of adaptive supply voltage and body bias for reducing impact of parameter variations in low power and high performance microprocessors. *IEEE Journal of Solid-State Circuits*. 2003;38(5):826-9. 10.1109/JSSC.2003.810053.
32. Tiwari A, Sarangi SR, Torrellas J. ReCycle: Pipeline adaptation to tolerate process variation. *ACM SIGARCH Computer Architecture News*. 2007;35(2):323-34. 10.1145/1273440.1250703.
33. Ghosh S, Bhunia S, Roy K, editors. A new paradigm for low-power, variation-tolerant circuit synthesis using critical path isolation. *Proceedings of the 2006 IEEE/ACM international conference on Computer-aided design*; 2006. 10.1145/1233501.1233628.
34. Gupta P, Pandey R. Dual output voltage differencing buffered amplifier based active-C multiphase sinusoidal oscillator. *International Journal of Engineering*. 2021;34(6):1438-44. 10.5829/ije.2021.34.06c.07.
35. Katebi M, Nasri A, Toofan S, Zolfkhani H. A temperature compensation voltage controlled oscillator using a complementary to absolute temperature voltage reference. *International Journal of Engineering, Transaction B: Applications*. 2019;32(3):710-9. 10.5829/ije.2019.32.05b.13.
36. Tasiran S, Demir A, editors. Smart Monte Carlo for yield estimation. *Proc ACM/IEEE TAU*; 2006.
37. Veetil V, Chopra K, Blaauw D, Sylvester D. Fast statistical static timing analysis using smart Monte Carlo techniques. *IEEE*

- Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2011;30(6):852-65. 10.1109/TCAD.2011.2108030.
38. Thanh BD, Le TH, Quoc VD. A Comparative Analysis of Axial and Radial Forces in Windings of Amorphous Core Transformers. *International Journal of Engineering*. 2024 Jan 1;37(1):201-12. 10.5829/ije.2024.37.01a.18
 39. Dey A, Biswas S. Shot-ViT: Cricket Batting Shots Classification with Vision Transformer Network. *International Journal of Engineering*. 2024 May 9. 10.5829/ije.2024.37.12c.04.
 40. Haghzad Klidbary S, Javadian M. Improvement of Low Energy Adaptive Clustering Hierarchical Protocol Based on Genetic Algorithm to Increase Network Lifetime of Wireless Sensor Network. *International Journal of Engineering*. 2024 Sep 1;37(9):1800-11. 10.5829/ije.2024.37.09c.10.
 41. Mirzaei M, Mohammadi S. Low-power and variation-aware approximate arithmetic units for Image Processing Applications. *AEU-International Journal of Electronics and Communications*. 2021;138:153825. 10.1016/j.aee.2021.153825.
 42. Chatterjee S, Kishinevsky M, Ogras UY, editors. Quick formal modeling of communication fabrics to enable verification. 2010 IEEE International High Level Design Validation and Test Workshop (HLDVT); 2010: IEEE. 10.1109/HLDVT.2010.5496662.
 43. Chatterjee S, Kishinevsky M, Ogras UY. xMAS: Quick formal modeling of communication fabrics to enable verification. *IEEE Design & Test of Computers*. 2012;29(3):80-8. 10.1109/MDT.2012.2205998.
 44. Sanders WH, Meyer JF, editors. Stochastic activity networks: formal definitions and concepts*. School organized by the European Educational Forum; 2000: Springer. 10.1007/3-540-44667-2_9.
 45. Deavours DD, Clark G, Courtney T, Daly D, Derisavi S, Doyle JM, et al. The Mobius framework and its implementation. *IEEE Transactions on Software Engineering*. 2002;28(10):956-69. 10.1109/TSE.2002.1041052.
 46. Sokolov D, Khomenko V, Mokhov A. Workcraft: Ten years later. This asynchronous world Essays dedicated to Alex Yakovlev on the occasion of his 60th birthday. 2016:269-93.
 47. Workcraft Website 2022. Available from: <https://workcraft.org/>.
 48. Mirzaei M, Mosaffa M, Mohammadi S. Variation-aware approaches with power improvement in digital circuits. *Integration*. 2015;48:83-100. 10.1016/j.vlsi.2014.07.001.
 49. Adl SMT, Mirzaei M, Mohammadi S. Elastic buffer evaluation for link pipelining under process variation. *IET Circuits, Devices & Systems*. 2018;12(5):645-54. 10.1049/iet-cds.2017.0394.
 50. Paliwal WD, Shastry P, Dighade S, editors. High performance using synchronous elastic circuits with lower overheads. 2014 International Conference on Advances in Electronics Computers and Communications; 2014: IEEE. 10.1109/ICAEECC.2014.7002453.

Persian Abstract

چکیده

این مقاله با بهره‌گیری از مدل‌سازی سطح بالا و اطلاعات تأخیرهای واقعی، چهارچوبی ارائه داده است تا بتوان به کمک آن، انواع طراحی‌های کشسان معمولی و ارزیابی زودهنگام را مدل نمود و به طور هم‌زمان آنها را از نظر درستی و کارایی در حضور تغییرپذیری با دقت بالا ارزیابی نمود. بررسی دقیق توان، تأخیر و PDP نشان می‌دهد که اجزای مربوط به ارزیابی زودهنگام نسبت به تغییرپذیری فرایندی حساسیت بیشتری نسبت به اجزای معمولی دارند. به‌ویژه، نتایج مدل‌سازی ریزپردازنده کشسان DLX کاهش ۲۶ درصدی در گذردهی و خطر ۰.۲ درصدی برای خطاهای همزمانی جریان داده و جریان کنترلی شبکه‌های کشسان را نشان می‌دهد که تأثیر قابل‌توجه تغییرپذیری را برجسته می‌کند.
